IntechOpen

Underwater Vehicles

Edited by Alexander V. Inzartsev

Underwater Vehicles

Edited by
Alexander V. Inzartsev

Underwater Vehicles
http://dx.doi.org/10.5772/107
Edited by Alexander V. Inzartsev

Notice

Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

First published in Croatia, 2009 by INTECH d.o.o.
eBook (PDF) Published by IN TECH d.o.o.
Place and year of publication of eBook (PDF): Rijeka, 2019.
IntechOpen is the global imprint of IN TECH d.o.o.
Printed in Croatia

Legal deposit, Croatia: National and University Library in Zagreb

Additional hard and PDF copies can be obtained from orders@intechopen.com

Underwater Vehicles
Edited by Alexander V. Inzartsev

p. cm.

ISBN 978-953-7619-49-7

eBook (PDF) ISBN 978-953-51-5743-4

Preface

The book offered to your attention, is dedicated to unmanned underwater vehicles (UUV). The UUV family is known to have two separate branches: the Remotely Operated Vehicles (ROV) and Autonomous Underwater Vehicles (AUV). Each branch has its advantages and limitations, and specific tasks. The difference between AUVs and ROVs is that AUVs employ “intelligence”, such as sensing and automatic decision making. They have predefined plan of operations in its “mind” allowing them to perform tasks autonomously. ROVs are controlled remotely by a human with the help of communication links on the basis of tether (cable, fiber optic, etc). However, the application of AUV technology to ROVs (transformation them into “smart” ROVs) is decreasing the differences between the branches. Originally there was the word "intelligent" at the book's title, which as it seems to me, correctly reveals tendencies of UUVs evolution. So, AUVs are the main theme of the most articles included in the book.

For the latest two - three decades in the various countries having leading position in the sea technologies, the significant number of AUVs has been created for the solving of wide spectrum of scientific and applied tasks of ocean development and research. For the short time period the AUVs have shown the efficiency at performance of complex search and inspection works and opened a number of new important applications. Initially the information about AUVs had mainly review-advertising character but now more attention is paid to practical achievements, problems and systems technologies. AUVs are losing their prototype status and have become a fully operational, reliable and effective tool.

Modern multi-purpose AUVs represent the new class of underwater robotic objects with inherent tasks and practical applications, particular features of technology, systems structure and functional properties. The problems connected with AUVs creation and development are versatile and in many cases do not have the complete solutions yet. Expansion of vehicles functionalities is also associated with the solution of some new theoretical problems. These are problems of control and navigation, orientation in underwater space, "intellectualization" of vehicles behavior, gathering and accumulation of the various environmental information and, at last, AUVs safety in design modes and in critical situations.

AUV creation belongs to number of the most priority tendencies in sea technologies. One of driving motives in the development of AUVs is human safety in underwater operations. Experience of the United States, Canada, Japan and European countries indicates capability of autonomous vehicles to solve the wide range of problems. Some examples of AUV applications include bathymetric and environmental mapping, marine geological survey, under-

ice deployment, pipeline and cables tracking, inspection and light intervention tasks, ecological monitoring, and a number of other applications including military ones. These different applications cover a variety of AUVs, from handheld gliders to large and heavy ones, and although AUVs are not the solution to all underwater operations they can offer a great increase in effectiveness.

The materials of the book cover the basic problems, development tendencies and application scopes of AUVs, namely:

- navigation problems;
- motion control methods and dynamics questions;
- AUV structure (including, mechanics, hard- & software);
- inspection of underwater communications;
- obstacle avoidance technique;
- ecological monitoring and pollution localization;
- automatic AUV docking;
- bio-robotic (fish-like vehicles);
- gliders.

I think, the book will allow the readers to get some ideas about modern underwater vehicles. Enjoy your read!

Alexander V. Inzartsev
Head of control systems lab.
the Institute of Marine Technology Problems FEB RAS
Far Eastern Branch of the Russian Academy of Sciences
Russia

Contents

1

Some Issues on the Design of a Low-Cost Autonomous Underwater Vehicle with an Intelligent Dynamic Mission Planner for Pipeline and Cable Tracking

Gerardo Gabriel Acosta[1-2], Hugo Curti[3],
Oscar Calvo Ibáñez[4] and Silvano Rossi[1]
[1]*Grupo INTELYMEC – Univ. Nac. del Centro de la Prov. de Buenos Aires.*
[2]*Consejo Nacional de Investigaciones Científicas y Técnicas CONICET.*
[3]*Grupo INTIA - Univ. Nac. del Centro de la Prov. de Buenos Aires.*
[4]*Grupo de Tecnología Electrónica – Univ. de las Islas Baleares.*
[1,2,3]*Argentina*
[4]*España*

1. Introduction

The main goal of this chapter is to introduce to the design and working hypotheses for the construction of a low-cost Autonomous Underwater Vehicle (AUV) devoted to pipeline and cable inspections. Underwater inspection is mainly necessary on account of the periodic surveys for the preventive maintenance of submarine infrastructure in the off-shore industries. The advantages of doing them with AUVs instead of Remote Operated Vehicles (ROV) or Towed Unmanned Devices (TUD) are low costs and better data quality in the inspection missions. The vehicle presented in this chapter was thought as an experimental platform to test multiple algorithms and to develop new technology, mainly using artificial intelligence techniques. In this sense, its modular architecture tackles the four main needs that should be solved in an autonomous vehicle to perform useful tasks in an unknown and strongly disturbing environment like the underwater world. They are: a) a robust **control system** to manage nonlinearities and disturbances, b) a precise **guidance system** to avoid unnecessary time and thus energy consumption, c) an accurate **navigation system** to determine self and target's positions, and d) an intelligent **dynamic planner** proposing the best possible trajectories and actions to successfully reach the mission objectives, based on decisions taken without human intervention. In a comparison with a biological being, the autonomous robot also need some kind of controlled muscles and forces to move, self perception and notion of the surroundings, and a brain to plan actions and movements.
In this work, different approaches for all of the aforementioned systems will be presented and thoroughly analyzed at the light of experimental evidence and author's experience in mobile robots (Fernández León et al., 2008). These experiments comprise computer simulations, hardware in the loop simulation as well as sea trials with the low-cost prototype described in the sequel, which is expected to navigate in the sea up to 100m of depth.

This chapter is organized as follows. A brief introduction to the application problem and a context for autonomous underwater vehicles will be presented in section 2. The hardware and software architecture for the AUV prototype will be discussed in sections 3. Section 4 will be devoted to experimental results analysis, and final conclusions of the whole chapter and future works will be given in the final sections.

2. Autonomous underwater vehicles for pipeline and cable inspections

2.1 Overview

The current need of energy transport such as electricity, petroleum and gas have provoked an increasing amount of underwater infrastructure such as cables and pipelines. In order to maintain these infrastructures with a suitable degree of safety and reliability, periodic inspections for preventive maintenance are necessary.

Damages in the submarine pipelines due to suspended (free-span) sections, craft anchorages or fishing activities can cause a strong environmental impact and to cut off certain critical supplies or communication lines. Although leakages of a submarine pipeline are not frequent, the consequences of an eventual spillage to the environment may be severe and irreversible. On the other hand, a greater demand of inspections and preventive maintenance are necessary due to pipeline ageing. In addition, many parts of the geographical areas with submarine infrastructure are located in deep waters (500-3500m.), constituting another challenge for the current technology.

A similar situation is observed in the case of preventive maintenance of submarine cables, because there is not an international standard to carry out it. A great reason to introduce these periodic surveys for the preventive maintenance is to reduce the repair time and, therefore, the profit losses due to the impossibility of information transmission through the cable. In addition, the spillages of pollutants of the damaged electrical submarine cables have also a dramatic impact in the fragile marine environment. To minimize it, it is necessary to urgently locate any possible damage, in order to take the necessary precautions for avoiding the pollution. Therefore, the above-mentioned maintenance also includes the recognition (and the corresponding decision about navigation behaviour of the submarine robot) of wastes located in the proximity of the inspected object. Thus, the shape recognition of fishing nets, rocks, mines, anchors, and other debris, should be also considered.

There are then two main motivations for preventive maintenance: to avoid infrastructure damage and for ecosystem preservation, which are closely related. Based on the previous observations, it is clear that one of the most outstanding applications for AUVs is pipeline and cable tracking for maintenance purposes. This explains the increasing interest on commercial exploitation of periodical underwater inspections.

Currently these inspections are done with ROV or TUD as mentioned, but these approaches have two basic drawbacks when compared to an AUV without a physical link to the surface: a) the lower quality of acquired data due to umbilical perturbation over position, and b) the higher cost to be invested in the ship and its crew each time that an inspection has to be undertaken. These two unwanted characteristics become enhanced, as the surveys depths are greater. For instance, offshore petroleum exploitation is being shifted to deeper waters as the resource is becoming scant. In contrast, AUVs allow a smother and faster navigation (over the typical three quarter knots of ROVs), and then a more reliable data acquisition is obtained. In fact, an AUV can reach positions in global coordinates and navigate in autonomous way with low position error, and is able to follow certain sensors readings

considering them for planning the desired and possible trajectory to be tracked the vehicle's control systems. However, there are other limitations like the pressure that the submarine can stand when the depth increases and the endurance in terms of battery power. These are still open problems from standpoint of new materials.
During the last years successful trials have been done with AUVs applied to cable and pipeline tracking. Among them, the paradigmatic Twin-burger 2, guided by cam images (Balasuriya & Ura, 1998), although for deep and opaque waters it is preferable to use sonar or a fusion of many sensors like in RAIS (Antonelli et al., 2001). Also the EU funded AUTOTRACKER Project, in which the authors participated, was thought to show that the current technology is mature enough to face this autonomous underwater pipelines and cables inspections in deep water up to three thousand meters. Some reports on preliminary successful results may be found in (Evans et al., 2003) and (Acosta et al., 2005). They were the antecedents for the AUVI prototype, also supported by the EU and the University of Balearic Islands, (Acosta et al., 2006), and (Acosta et al., 2007). The AUVI was constructed mainly to test computational intelligence algorithms for planning and replannig of vehicle's trajectories and tasks, and is the ancestor of the current prototype ICTIOBOT.

2.2 AUV general architecture for target tracking

The necessary building blocks for an AUV devoted to pipeline and cable tracking are depicted in figure 1, and explained in the following paragraphs.

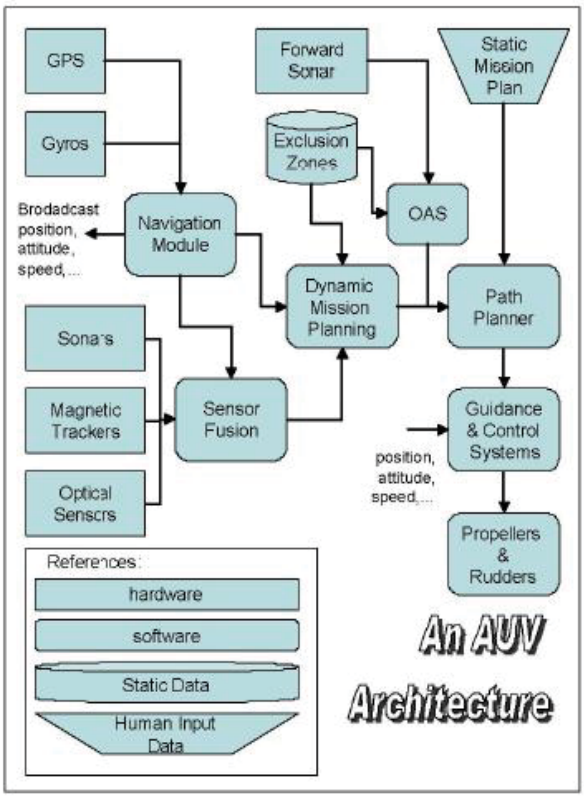


Fig. 1. Building blocks for an AUV software/hardware architecture

The **navigation module** is usually referred to the on board sensory systems. It comprises the data fusion necessary to locate precisely in 3D the AUV rigid body and the target position. The usual components within the navigation system are a global position system (GPS), an inertial navigation system (INS), a compass, a depth sensor, and others. Thus, the navigation system provides the dynamic mission planner system, the guidance system and the control system with accurate data to achieve their objectives.
The **guidance module** is frequently associated to a low-level trajectory generation. When the waypoints for the robot are defined, a trajectory to reach them is necessary in order to feed the controllers set points.
The **control module** is regarded to the feedback loops allowing the vehicle to describe the trajectory as close as possible to the proposed path given by the guidance module. In effect, assuming that the navigation system yields a clear perception of the AUV's positions, speeds and headings, and the guidance system gives a suitable trajectory to reach a waypoint, there is still remaining a module capable of maintaining the vehicle as close as possible to the prefixed trajectory. Established in this way, the problem to solve at this stage is a control problem to command the vehicle actuators (propellers, rudders and pumps).
A **mission planner**, according to the robot's application, is also necessary to accomplish the task in an autonomous way. A key component of the mission plan is the path planning. Special sensor acquisitions (snapshots, videos, water samples, and others) or special actions (debris grasping) may also be considered within a mission planner. The mission plan consists of two kinds of objectives: long term static objectives and short term changing ones. The first ones, are given beforehand, in a rigid way through a human-machine interface, and then conform a **Static Mission Planning**, (SMP). The short term objectives can be changed on-line, and constitute the **Dynamic Mission Planning** (DMP), and varies as the vehicle movement progresses in the real world. This is also known as mission replanning, in response, for instance, to different obstacles to be avoided, based on data from a forward looking sonar and the pipeline or cable position. The artificial intelligence based DMP is a core development in the AUV presented in this article, so it will be explained in more detail in a following section.
When the underwater installation is constructed, legacy data (LD) are archived containing pipeline's or cable's positions, depths, a corridor width and forbidden zones. They should be on-line accessed by the DMP and the obstacle avoidance system (OAS).
Pipeline trajectory may be estimated from special sensors like a multi-beam echo sounder (MBE), a side-scan sonar (SSS), a magnetic tracker (MAG), a DIDSON sonar (DID), cameras (CAM) and others. This information is combined in a **sensor fusion module** (SFM) yielding a position and direction estimate of the target to be inspected. From these data, the DMP is able to decide a trajectory according to different situations like searching a pipeline, following it, navigating closer to it, or recognizing other objects surrounding it. This desired trajectory is defined as a collection of four (4) waypoints to be reached by the vehicle.
The **Path Planner** in this architecture only decides if the desired trajectory given by the DMP is possible or not, according to the outcome of the OAS. Then the waypoints belonging to the desired and possible trajectory are inputs to the guidance module.
The **Obstacle Avoidance System** (OAS) receives data from a forward-looking sonar (FLS). When an obstacle is detected, a near possible waypoint is proposed to correct the desired trajectory from the DMP.

2.3 An overview of tracking methods

The most common methods to track submarine infrastructures are based on magnetic and electromagnetic principles. Robots like ROVs or remote operated towed vehicles are capable of transmitting signals in such a way that the operator takes advantage of them to control its movement by means of a joystick.

In addition to magnetic tracking methods, the operator can use a video camera to prevent seabed collisions. Nevertheless, this method is limited by the poor visibility of the water and can be improved by means of obstacles detection on the basis of acoustic image systems, as already tested in the AUTOTRACKER project (Evans et al., 2003), (Acosta et al., 2005).

The movement estimation and pattern shape recognition from video sequences have received a great attention on behalf of the computer vision community, and several solutions for calibrated and uncalibrated cameras have been proposed in the last years. The current methods use estimations based on optical flow techniques (Barron et al., 1994), that is, the instantaneous velocity field of every image pixel, or in characteristics trackers (Shi & Tomasi, 1994), who resort to algorithms looking for special elements of the image and they follow them through the whole sequence.

Some existing methods for characteristics tracking seem to be sufficiently solid and mature to support movement estimation and shape recognition in a reliable manner, without an intrinsic computational load associated with optical flow techniques (Rives et al., 1986), (Looney, 1997). However, still it is necessary to investigate how to adapt the existing solutions to the underwater images, since many of the suppositions commonly assumed in the air are violated. For instance, the lightning is not static and independent from the observer movement.

Thus, sonar data processing research has focused in the objects detection, its classification, the obstacle avoidance, and the navigation based on the terrain. Forward-looking sonars (FLS) with mechanical sweep provide of richer information, but they need the correction of the movement using navigation information from the vehicle that transports it. The multibeam MBEs are bigger than the FLS and capable of providing several updates of the image frames. From MBE and by means of simple methods of image processing it is possible to extract useful information of the pipeline or cable for its tracking. Nevertheless, the great challenge is still to reduce the false alarms relation using multiple hypotheses between frame and frame for the tracking.

There is another open problem related to the capability of detecting and classifying different objects from those that are under inspection, employing the same information from the sonars. Likewise, when the objective under study gets lost, the search strategies and reacquisition can be optimized to minimize the search time of the beginning of mission point, or the location, in the future, of targets of different types and shapes. For instance, applying variants of "Random Walk" and "Lévy Fligth" models (Bartumeus et al., 2002).

Both the FLS and the video sensors are commonly available in the submarine commercial vehicles, but they are used by the human pilots as navigation and decision sensors. For its utilization in an AUV, this sensor data must be processed on-line and in real time, and used to provide the input to the control systems in order to carry out a correct tracking and obstacle avoidance. Though there are laboratory systems that do this currently, it must be carefully demonstrated the practical validation in a real system working in the seabed. The state of the art has been demonstrated inside the European programs Esprit III and MAST III, in which ROVs were used with general-purpose computers calculating in the surface.

The results of these programs have been promissory, indicating that would be opportune to consider fixing these systems directly in the vehicle, which would be obviously an AUV.

3. The low-cost prototype ICTIOBOT

The general objective of the work described in this chapter was to design and to develop cost-efficient technology for the inspection of pipelines and submarine cables in maritime infrastructures, including the development of an autonomous and safe navigation system for a submarine prototype. With this prototype it will also be possible to make pattern recognition from sonar's data to detect debris in the proximity of the target to be tracked.
The design considerations and proposed solutions as well as a description of the hardware employed, the sensors in the payload, and the DMP based on an expert system are presented in this section.
The specific aims of this development were the following ones:

- To develop a software module to provide the desired trajectory for an AUV devoted to submarine pipeline and cable tracking with the purpose of inspecting them, resorting to artificial intelligence techniques, particularly knowledge based systems (KBS) and artificial neural networks (ANN). The initial point was the EN4AUV (Expert Navigation for Autonomous Underwater Vehicle) in (Acosta et al., 2003). This module will be more thoroughly described in section 3.5.
- To assemble a low-cost prototype, including a MBE sonar, a SSS sonar, a GPS, an INS, and an industrial PC type as processing unit.
- To develop a pattern recognition system based on artificial neural networks for classifying objects using the data provided by the sonars.
- To validate the resulting prototype by means of its utilization in the sea for carrying out inspections of pipelines and submarine cables to a depth not bigger than 100m.

3.1 Working hypotheses

A modular philosophy was employed in the design and development of the low-cost AUV prototype because several modules that were previously tested in other applications, were reused here. In particular, there is a great interest on knowledge reuse through the generalization in an ontology for mobile robots navigations. In this manner, the design of a knowledge-based DMP easily ported from one application to another is an essential point to research and test within this undergoing project. To accomplish this goal, some working hypotheses were considered:

- The hardware technology is able to deal with pipeline and cable tracking in submarine environments.
- The KBS approach is sufficiently robust for managing the decision process in the real-time planning and replannig of the better trajectory in complex environments.
- It is possible to detect and to recognize different shapes in the seabed from acoustic images by means of the utilization of ANN, and this is much more efficient than using an optical camera for the dark waters of the Argentinean Sea.
- The whole technology and the knowledge base already developed in previous projects can be reused in the ICTIOBOT prototype, for researching in planning, guiding, controlling and navigating algorithms, mainly based in computational intelligence approaches.

- The prototype should be of low cost not only regarding the effective cost of the prototype but also the operations costs of each sea trial.

3.2 ICTIOBOT's architecture

The modular architecture of the prototype is based on the aims and working hypotheses previously described, and it is presented in figure 2, showing the basic modules of any autonomous robot, that is, the mission planner planner, guidance, control, and navigation systems.

The DMP system comprises the EN4AUV knowledge based system and an OAS, based on evolutionary robotic paradigm developed for terrestrial applications (Fernández León, et al. 2004). Both modules, jointly with the guidance and control systems, and the pattern recognition system run in a low consumption on-board PC.

The navigation system is constituted by a compass, a GPS (for navigating in the surface and re-initialization of accumulated errors), an inertial navigation system, and a depth sensor. They are the input of a sensor fusion module that determines the position in absolute coordinates, the attitude and the depth of the robot location.

The other sensors in the payload are a MBE for target tracking and a FLS for obstacle perception. A SSS and a MAG for target searching and tracking will be added in future stages of the project.

Every module has an input-output data flow based on messages put on a UDP channel, and broadcasted for the remaining ones, using TCP/IP protocol. The operating system is GNU Linux running in an industrial PC, and the preponderant programming language is C++.

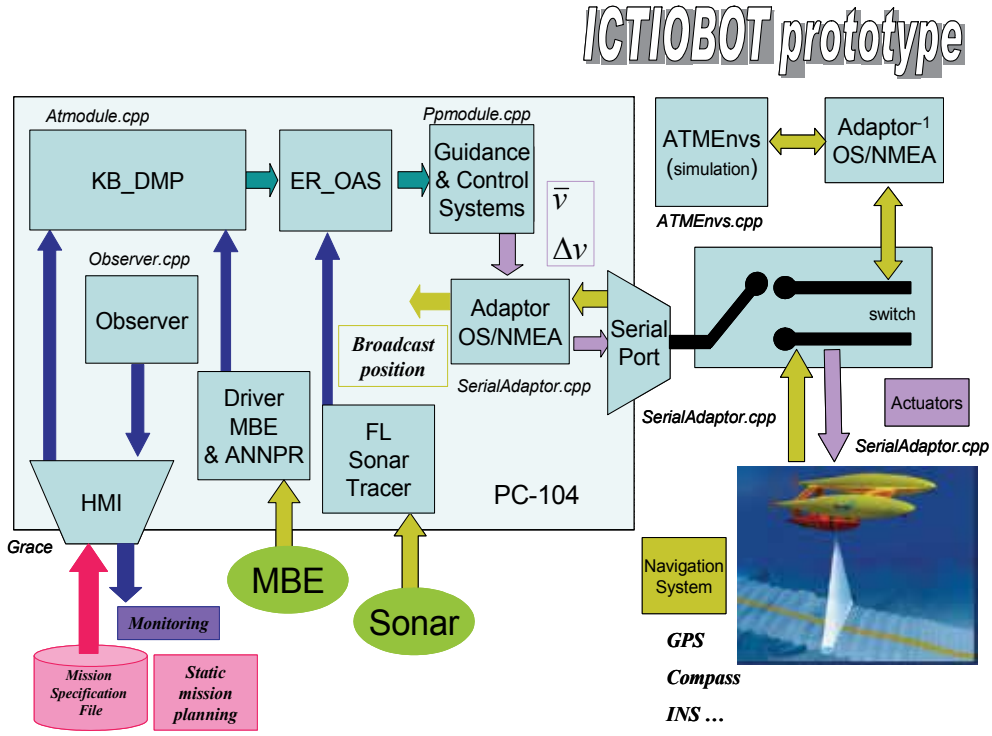


Fig. 2. Modular architecture for the ICTIOBOT prototype

3.3 Hardware architecture

The AUV hardware is composed of two main processor boards: a low level and a high level electronics. The low level electronic is based on the iCm4011 microcontroller development board from Ingenia that uses a dsPIC 30F4011 as main processor. This board provides I2C and RS232 interfaces for communicating with the sensors and the high level processor unit. The high level board is an industrial PC based on Intel X86 microprocessor. This is the environment where most of the planning and controlling software resides. Figure 3 shows, in schematic way, the on-board electronics for the low-coast AUV prototype.

The microcontroller board gets the position either from the GPS when surfaced, or from an accelerometer, integrating both x and y readings. From this data and the reading data provided by the depth meter, the microcontroller board is capable to provide the (x, y, z) triplet necessary for the proper position. The GPS sends its data using RS232 streams according to a format specified by the National Marine Electronics Association (NMEA). The accelerometer is based on the Analog Devices ADXL 202, dual axis accelerometer, constituting an inexpensive device that measures accelerations with a full-scale range of ±g. This sensor outputs are analog voltages or digital signals whose duty cycles, that is, the ratio of pulse width to period, are proportional to acceleration. The duty cycle outputs can be directly measured by the microprocessor.

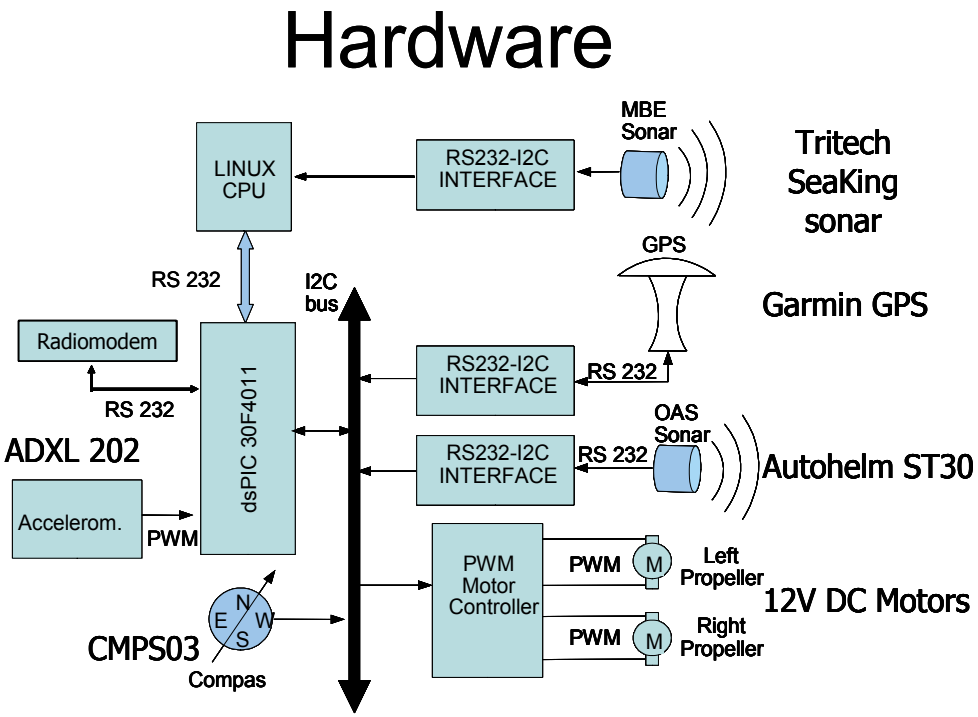


Fig. 3. The ICTIOBOT on-board electronics.

On the subject of the target tracking sensors, the MBE sonar is a Tritech Seaking and converts it readings into RS232 streams according to a proprietary data format. The obstacle avoidance sonar is an Autohelm ST30 depth finder and sends its echoes using Seatalk protocol (NMEA) by RS422 electrical signals that can be easily converted into RS232 signals

Both the low level processor board and the high level board also communicate through RS232. Given the wide number of RS232 channels needed to interface all the modules and navigation sensors, it was decided to turn some of these signals into I2C communication interface format. This popular and versatile bus used for communications among devices nodes inside equipments, can communicate with the slaves using different addresses and is easily configurable.
The digital compass is a CMPS03 robotics board based on the Philips KMZ51 magnetic field sensor, and is used to obtain the AUV orientation with respect to the earth magnetic field. This device, directly interfaced with the PC, provides a pulse width modulation (PWM) signal with the positive width of the pulse representing the angle apart of the north point.
As regards the actuators, two propellers provide the horizontal movement with no need of rudder, and its DC motors are driven by a MD22 Dual H-Bridge device made by Robot Electronics capable to provide 50V- 5A. The MD22 driver is based on a PIC microcontroller and accepts data communication over I2C bus. The vertical movement is provided by two propellants controlled by a MD03 50V – 5A driver from Devantech.
Finally, a link radio provides a safety mechanism to handle manually the AUV in the event of possible damages or flaws.
Any signal that comes from the transmitter at the frequency of 422MHz is considered as an interruption input to the microcontroller of the low level board, and consequently, stops attending the orders from the PC's communication serial port.

3.4 Mechanical parts

The low-cost ICTIOBOT prototype presented in figure 4 consists of two torpedoes assembled in an aluminium structure. These torpedoes are stimulated by two electrical propellants constructed by Motorguide and used by the divers as dragger vehicles. These are economic, support pressures of up to 6kg and develop a speed of up to 2mph each one. They are done of glass fibber and take 4 batteries inside. The above mentioned batteries are of 12V 33AH of absorbed electrolyte that give a good autonomy, considering that the tests of bigger duration and exigency reached 4 hours.
A high pressure canister is placed between both torpedoes and contains most of the mentioned electronics, except the compass and the communication systems, which are assembled apart in a watertight plastic box over the aluminium structure. The canister in turn is connected to other PVC box, everything using cable connectors that support pressure values associated with depths over 100m. The GPS, the high frequency modem and the WiFi adapter are protected by a water-repellent gel inside a few PVC's pipes.
The MBE is assembled in the front of the AUV structure and protected by a steel mask of a few aluminium bars that allow its vertical or horizontal assembly.

3.5 Expert system based dynamic planner

During an inspection, several unforeseen situations might appear like the detection by the FLS of a fishing net, or a complex pattern shape of more than one pipeline over the seabed recognized by the MBE, or simply a detour due to obstacle detection. In these cases, it is hoped that the DMP module exhibits an "intelligent" behavior. To cope with these real situations in the marine world, it was resorted to the experience and skills of ROV operators. A little part of their knowledge was elicited and codified in the form of a real time expert

Fig. 4. Photographs of the low-cost ICTIOBOT prototype

system, the EN4AUV introduced earlier. It has been developed using CLIPS that constitutes a C language based shell, and allows the knowledge representation to be in the form of rules and frames (COOL or Clips Object Oriented Language). These formalisms are used in the knowledge base (KB) to represent the involved knowledge. The main feature of the expert system is to assess a current situation in order to act accordingly, in a clearly data driven/reactive behavior. Thus, EN4AUV is a reactive expert system, taking the proper action for every different situation, and considering the pipeline/cable status, the type of survey, the different mission settings, and others.

These situations were coded as possible scenarios in about fifty rules, like the one presented in figure 5. As the knowledge about different situations increases, the knowledge base describing new scenarios can be completed an updated, yielding an incremental KB growth.

Each scenario triggers different searching or tracking strategies, which are then subtasks with their own features. Scenarios are based mainly in two ideas: a) the survey type, and b) tracking states. The first one is defined a priori in the SMP module, to establish the number of pipeline/cables to be tracked, the navigation depth, and other mission features. The other basic component of the scenario determination is the tracking state that changes when the SFM module updates its sensors. From this information, the EN4AUV is able to decide the status of the pipeline/cable, that is, if buried, exposed, intermittent or free-span, and how is the AUV as regard as the pipeline/cable, that is, if avoiding an obstacle, with the object under study considered as found or lost, or returning to a previous known position.

```
(defrule R05.1
    (R43D)
    ?a <- (R4SD)
    ?ws <- (object (is-a WORKING_SCENARIO)
(Count_Reacq ?cr)(Search_results ?sr) (Movie $?MOVIE)
(Follow_status ?FS))
    ?sv <- (object (is-a SURVEY) (Max_Reacquire ?mx))
    (test (<= ?cr ?mx))
    (test (eq ?FS LOST))
    (test (eq (send [OBJ_STUDY] get-
Present_Layout_Status) NOT_DETECTED_NOT_BURIED))
    test (= 0 ?sr))
=>
    (assert (Current_scenario SC8))
    (assert (PPLS notready))
    (retract ?ws)
    (insert$ $?MOVIE 1 SC8)
    (send ?ws put-Movie $?MOVIE)
    (retract ?a) ;to avoid the assertion of multiple scenarios in
the same KB query
    (printout t "CLIPSMACHINE: R05.1 Current Scenario is
SC8 -searching a pipe/trunkline" crlf)
    (printout t "CLIPSMACHINE: inserting movie" $?MOVIE
crlf))
```

Fig. 5. A rule from the knowledge-base of the DMP for the 8th scenario determination, in the typical CLIPS syntax

Once scenarios are established, a typical situation assessment task, EN4AUV must output a desired trajectory or must decide a pipeline/cable reacquisition.

To yield a desired trajectory, the actions are organized in a set o few simple subtasks: findstart, search, back to start, skip, and track. Then the final trajectory of the AUV is built by one of these subtasks, or by a concatenation of them. Examples of scenarios developed for the sea trials were the following ones:

- 1st Scenario: The AUV is tracking an exposed pipeline, navigating on top, at a fixed offset smaller or equal than 5 meters. Both the MBE and the MAG can detect it.
- 2nd Scenario: The AUV is tracking a buried pipeline on top, at a fixed offset, smaller or equal than 5 meters. The MBE may not be able to detect it, but the MAG can track it anyway.
- 3rd Scenario: The AUV is tracking an intermittently exposed and buried pipeline at a fixed offset. This is a sequence of alternative appearance of scenarios number one and two.

- 4th Scenario: The AUV is tracking a free-span pipeline at a fixed offset. The pipe is tracked mainly based on MBE readings, which may be detecting the pipe itself or the trench.
- 5th Scenario: The AUV is tracking a pipeline in the presence of one or more pipes (like infield pipelines) or other magnetic objects in the area. Measures from MBE as well as MAG are needed.
- 6th Scenario: The AUV is tracking a pipeline but avoiding an obstacle. In such scenario the certainty error may increase beyond its thresholds, but the EN4AUV knows where the pipe is and ignores the pipe_lost flag. The path planner module outputs a flag indicating this condition and the EN4AUV may query the legacy data to confirm the existence of an exclusion zone. Although sensor readings are not reliable, they are not turned off to be ready when the AUV is again over the pipeline.
- 7th Scenario: The AUV is searching a buried pipeline. No readings from MBE, just MAG will yield detection when the AUV is right over the pipe. With two detection (crossing) points the pipeline direction vector is computed and the AUV starts tracking from the last known point with this direction.
- 8th Scenario: The AUV is searching the pipeline, which is considered as lost. EN4AUV shall have an estimate of the trajectory from SFM considering the whole inputs: MBE, MAG, and LD. If not, it will estimate the initial parameters from mission settings and LD.
- 9th Scenario: The AUV is searching a pipeline in the presence of one or more pipes (like infield pipelines) or other magnetic objects in the area. Every information source is operative to discriminate the target under study (MBE, MAG, and LD).
- 10th Scenario: The AUV is skipping from one point to another. MBE, and MAG are off to save energy. This special situation appears when changing from one pipe to another to track, or from one zone of interest to another over the same pipeline.
- 11th Scenario: The AUV is going back to the last known position to start tracking, after founding the pipeline as a consequence of a successful search. MBE, and MAG are off.
- 12th Scenario: The maximum number of reacquisition after unsuccessful searches was reached. The mission is ended with a failure message.
- 13th Scenario: The AUV is tracking an exposed pipeline, navigating on top, at a fixed z_offset greater than 5 meters. The detection is done mainly with the MBE.
- 14th Scenario: The AUV is tracking a buried pipeline on top, at a fixed z_offset greater than 5 meters. The blind tracking is done mainly based on legacy data, and cannot last more than half a minute. After this, if there are no more sensor readings, a new search must be started.

3.6 Development and runtime environments

A leading idea was to employ the same environment for the software development than for the runtime, with a minimum adaptation time for porting it from one to the other. The subsystems were first tested in simulation, within a development environment named ATMEnvs (Curti et al., 2005) such that the software could be directly ported to the vehicle when these simulation tests were successful. Thus, the simulation environment was communicated with the vehicle software through the serial port, using an adaptor between the NMEA protocol and the messages within a proprietary communication protocol among modules. In this way, once the software is operative, the simulator is easily and

straightforward replaced by the real hardware prototype. The simulation environment included cinematic and dynamic linear models of the vehicle. The static information among the different modules was shared through bottom specification files (*.bsf) and objective under inspection specification files (*.psf). The information associated with the dynamic variables among the modules travel in the messages mentioned before, the same that is employed when the robot is operating in its real environment.
Referring to figure 2 in Section 3.2, the DMP and OAS constitute a system in cascade connection; consequently, if the OAS did not detect any object through the FLS, its output will be simply the desired trajectory from the DMP. On the contrary, if an obstacle is detected, the OAS changes the necessary waypoints in the trajectory provided by the DMP.

4. Experimental results

Many approaches were implemented and tested for the control system. For instance a linear PI-MIMO (Proportional Integrative – Multiple Input Multiple Output) plus feed-forward structure (Valenciaga et al., 2007) and also a nonlinear fuzzy logic based control were verified in computer simulation. They were applied in a low-cost ocean observing platform, based on a hybrid vehicle between an AUV and autonomous surface vehicle (ASV), called CORMORAN. This robot moves slightly underneath the sea surface following a previously planned route and regularly dives to make vertical profiles of the water column. The PI-MIMO control strategy is used to commands the rudder and the propeller.
However, the control system currently mounted in ICTIOBOT is a simple PI-MIMO control in the horizontal plane (2D) minimizing the angle β, that is given by Equation 1.

$$\beta = \lambda - \Psi \tag{1}$$

where Ψ is the vehicle's yaw and λ the angle given by Equation 2, where (x_1,y_1) and (x_2,y_2) the coordinates of the robot and the target waypoint, respectively (figure 6). This is known as the line-of-sight (LOS) guidance law (Naeem et al., 2003). The speed envelope using an ideal particle approach, recently proposed (Breivik and Fossen, 2005) was also tested in computer simulation (Valenciaga et al., 2007), but not yet ported to the prototype.

$$\lambda = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{2}$$

The navigation system for the preliminary trials showed in this chapter, consisted on a sensor fusion of measures from a compass, an inertial navigation system, a depth sensor and a GPS (when navigating in surface), for self position perception. For pipeline tracking, the prototype used an MBE, to avoid the need of light for image processing near the dark seabed when water is not clear or the applications shift to deeper waters.
The knowledge-based DMP was also integrated to the ICTIOBOT prototype. The evolutionary robotic OAS was not ready to be tested in these trials. Several tests were carried out in the sea surface and slightly underneath the surface, to have available the signal from GPS in all the cases.
These modules were tested in the trials done in Cala Estancia that is bay with scanty meters of depth, near to Palma de Mallorca, Spain, in April 2008. A curved trajectory was the reference of the step points, represented by solid red line in figure 7, whereas the punctuated line in the same figure represents the real position described by the AUV.

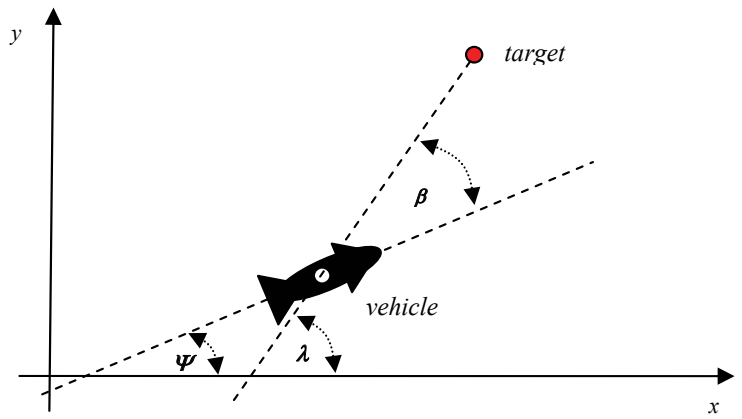


Fig. 6. Line of sight (LOS) guidance approach.

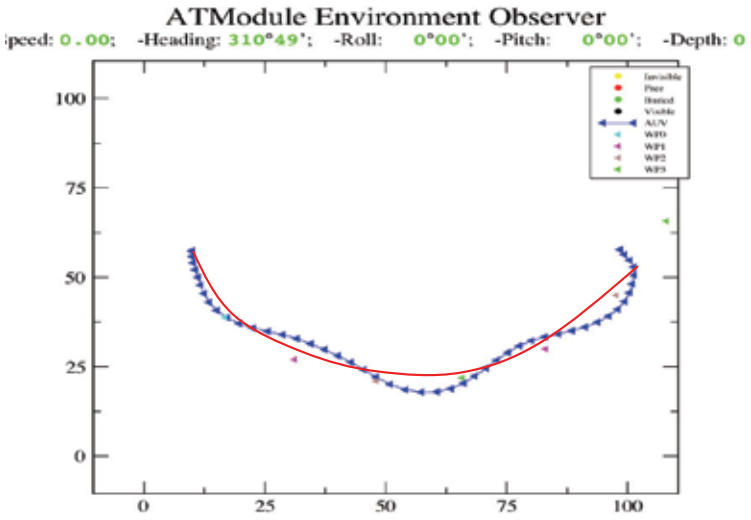


Fig. 7. Test of the ICTIOBOT prototype, in slightly deep waters of the coast of Mallorca, Spain, tracking a curved trajectory

As it can be seen, navigation, guidance and control systems are very elementary and even in these calm waters the trajectory presents severe deviations from the reference.

A second set of trials was carried out with a pipe in the seabed, painted in yellow in figure 8. As it may be seen, the prototype begins in the position (58.0), relative to the trial location, and moves to the position (65.25) to initiate the search. When this point is found, after doing a trajectory in shape of eight, the sonar is turned off in order to simulate a search strategy proposed by the DMP.

Thus the reference appears in zigzag for the wished path. This trajectory is depicted by step points which were united by a solid red line for better clarity. Again, the necessary improvement is demonstrated in the guidance and control systems, since with the bigger inertia on account of the weight of the sonar and the associate electronics, the AUV moves away from the reference imposed by the DMP.

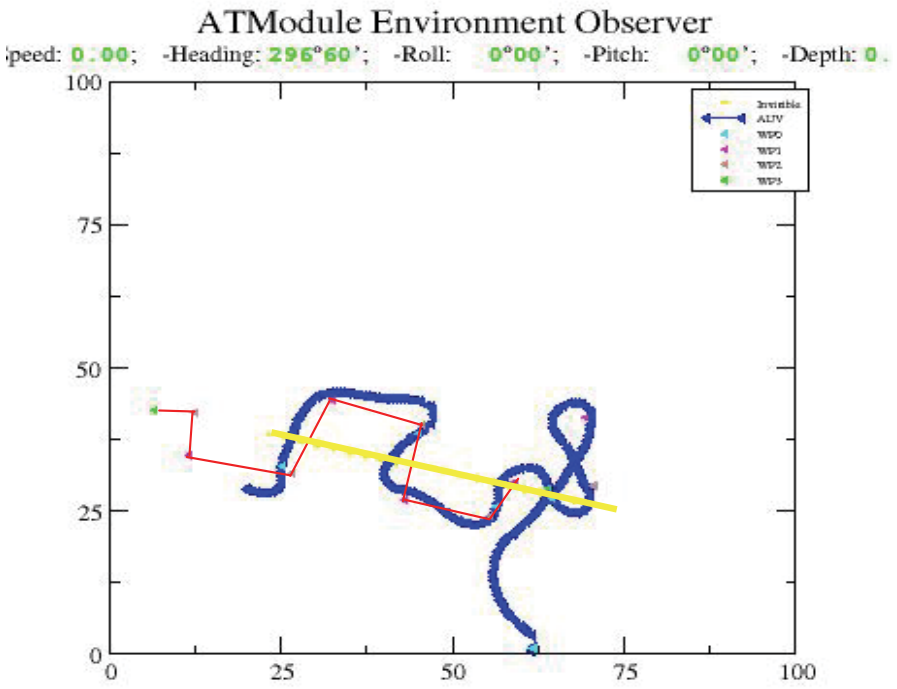


Fig. 8. Test in Mallorca, Spain; looking for a pipeline

5. Conclusion

The key aspects of the software and hardware elements design for an AUV prototype have been presented, as a low purchasing and operation cost adaptation of other vehicles over which authors have been working in other projects. In particular, the dynamic mission planner based on an artificial intelligence paradigm showed to be very suitable to be reused in different robots with different inspection missions. This mission planner showed that this approach is able to re-plan the vehicle trajectory while in the mission, taking into account the original static mission settings, the changing underwater environment and the situation of the target under inspection. The new experimental prototype ICTIOBOT presented here will surely be an adequate test-bed for the new task and path planning algorithms. In particular we will focus in a near future in the enlargement of the knowledge-base within the DMP for untested use-cases.

The work also shows that it is possible with current technology to construct a low-cost AUV for pipeline and cable tracking that can be used for preventive maintenance purposes of submarine infrastructure. In addition, the prototype is useful for test and improvement of each module separately.

From the obtained results is possible to observe that there is still a need to improve several aspects in the guidance, control and navigation systems, since the AUV's real path moves away considerably from the desirable trajectory proposed by the DMP module.

6. Future work

It will be proposed different approaches for each of the described subsystems: the Dynamic Mission Planner, the Guidance System, the Control System and the Navigation System. They will be evaluated and verified on the experimental prototype.
For the DMP, it is expected to add new rules for describing more complex scenarios and cases required by the users as, for instance, multiple pipelines tracking, recognition of different pattern shapes like fishing nets, mines, anchors, and wastes nearby to the pipeline. To achieve this it is proposed a devoted sensor that uses the sonar information for filtering it by means of an artificial neural network and operates as a special input to the knowledge base. For the Guidance and Control Systems, it is expected to continue searching which is the best approach for the problem to be resolved. Specially, is desirable that the system that finally controls the AUV is of the adaptive type in order to minimize possible errors that can appear due to on-board variable payload. For the Navigation System, after studies and experimental tests it will be selected a method to implement a sensor fusion system to determine the proper AUV position and attitude, with a bounded accuracy. Based on this information it will be possible to obtain the target positions and recognized patterns in the seabed. Among the possible approaches to be analyzed, the most promising seems to be Kalman's filters and those who combine sensor inputs using fuzzy logic techniques. It is also expected that a side-scan sonar and a magnetic sensor for target tracking will be added in future stages of the project to enhance the possibilities of target and obstacle perception of the AUV.

7. Acknowledgments

The research and development work described in this chapter was possible due to the financial support of the following projects: TRA2006-13318 MEC (Spain), PCI2005-A7-0356 FEDER (EU), AUVI-MIF2-CT-2004-003027, FP6 (EU), and PAE 22696 National Agency ANPCyT (Argentina).

8. References

Acosta, G. G. ; Calvo Ibáñez, O. A. ; Curti, H. J. & Rozenfeld A. F. (2007). Low cost autonomous underwater vehicle for pipeline and cable inspection, *Proceedings of IEEE UT07/SSC07 – Underwater Technology 2007 and Workshop on Scientific Use of Submarine Cables & Related Technologies 2007,* Vol. 1, pp. 331-336, ISBN: 1-4244-1208-0/07, Tokyo, Japan, April 17-20 2007.

Acosta, G. G. ; Curti, H. J. ; Calvo, O. A. & Rossi, S. R. (2006). A knowledge-based approach for na AUV path planner development. *WSEAS Transactions on Systems,* Vol. 5, No. 6, June 2006, pp. 1417-1424, ISSN: 1109-2777.

Acosta, G. G. ; Curti, H. & Calvo, O. (2005). Autonomous underwater pipeline inspection in AUTOTRACKER PROJECT : the navigation module, *Proceedings of IEEE-OCEANS'05 Europe Conference,* Vol. 1, pp. 389-394, Brest, France, June 21-23 2005.

Acosta, G. G. ; Curti, H. ; Calvo, O. & Mochnacs, J. (2003). An Expert navigator for an autonomous underwater vehicle, *Proceedings of SADIO/Argentine Symposium on Artificial Intelligence ASAI'03,* in CD, Buenos Aires, Argentine, September 2003.

Antonelli, G.; Chiaverini, S.; Finotello, R. & Schiavon, R. (2001). Real-time path planning and obstacle avoidance for RAIS: an autonomous underwater vehicle. *IEEE Journal of Oceanic Engineering,* Vol. 26, No. 2, April 2001, pp. 216-227, ISSN: 0364-9059.

Balasuriya, A. & Ura, T. (1998). Autonomous target tracking by underwater robots based on vision, *Proceedings of IEEE International Symposium on Underwater Technology*, pp. 191-197, Tokyo-Japan.

Barron, J. L.; Fleet, D. & Beauchemin S. (1994). Performance of optical flow techniques. *Int. Journal of Computer Vision,* Vol. 12, No. 1, pp. 43-77, ISSN: 0920-5691.

Bartumeus, F.; Catalan, J.; Fulco, U. L.; Lyra, M. L. & Viswanathan G. M.. (2002). Optimizing the encounter rate in biological interactions: Lévy versus Brownian strategies. *Physical Review Letters,* Vol. 88, No. 9, March 2002, pp. 1-4, ISSN: 0031-9007.

Breivik, M. & Fossen, T. (2005). A unified concept for controlling a marine surface vessel through the entire speed envelope, *Proceedings of IEEE Mediterranean Conference on Control and Automation,* pp. 1518-1523, Limassol, Cyprus, June 27-29 2005.

Curti, H. ; Acosta, G. G. & Calvo, O. (2005). Autonomous underwater pipeline inspection in AUTOTRACKER PROJECT : the simulation module, *Proceedings of IEEE-OCEANS'05 Europe Conference,* Vol. 1, pp. 384-388, Brest, France, June 21-23 2005.

Evans, J. ; Petillot, Y. ; Redmond, P. ; Wilson, M. & Lane, D. (2003). AUTOTRACKER : AUV embedded control architecture for autonomous pipeline and cable tracking, *Proceedings of IEEE-OCEANS 2003,* Vol.5, pp. 2651-2658.

Fernández León, J. A. ; Acosta, G. G. ; Mayosky, M. A. & Calvo Ibáñez, O. (2008). A biologically inspired control based on behavioural coordination in evolutionary robotics, Chapter in: *Advancing Intelligence through Biological Process Applications,* Idea Group Inc., pp. 107- 129, ISBN: 978-1-59904-996-0.

Fernández León, J.; Tosini, M & Acosta, G. G. (2004). Evolutionary reactive behavior for mobile robots navigation, *Proceedings of 2004 IEEE Conference on Cybernetics and Intelligent Systems (CIS04)*, pp. 532-537, Singapore, December 1-3, 2004.

Looney, C. (1997). *Pattern Recognition Using Neural Networks,* Oxford University Press, ISBN: 0-19-507920-5, Oxford/NY.

Naeem, W. ; Suttom, R. ; Ahmad, S. & Burns, R. (2003). A review of guidance laws applicable to unmanned underwater vhicles. *The Journal of Navigation,* Vol. 56, pp. 15-29

Rives, P. ; Breuil, E. & Espiau, B. (1986). Recursive estimation of 3D features using optical flow and camera motion, *LAS,* pp. 522-532.

Shi, J. & Tomasi, C. (1994). Good features to track, *Proceedings of IEEE Computer Soc. Conference on Computer Vision and Pattern recognition (CVPR'94),* pp. 593-600, Seattle, USA, June 1994.

Valenciaga, F. ; Puleston, P. F. ; Acosta, G. G. & Calvo, O. (2007). Trajectory tracking of the Cormoran AUV based on a PI-MIMO approach, *Proceedings of IEEE-OCEANS'07 Europe Conference,* pp. 1-6, ISBN: 978-1-4244-0635-7, Aberdeen, Scotland, June 18-21 2007.

2

Optimal Control of Underactuated Underwater Vehicles with Single Actuator

Mehmet Selçuk Arslan, Naoto Fukushima and Ichiro Hagiwara
Tokyo Institute of Technology
Japan

1. Introduction

The research on underwater systems has gained an immense interest during the last decades with applications taken place in many fields such as exploration, investigation, repair, construction, etc. Hereby, control of underwater systems has emerged as a growing field of research. Underwater vehicles, in fact, accounted for 21% of the total number of service robots by the end of 2004, and are the most expensive class of service robots (UNECE/IFR, 2005). Typically, underwater vehicles can be divided into three underwater systems, namely, the manned submersibles, remotely operated vehicles (ROV) and autonomous underwater vehicles (AUV). ROVs and AUVs are mostly utilized in the oil and gas industries, and for scientific and military applications. AUVs, especially are of great importance due to their ability to navigate in abyssal zones without necessitating a tether that limits the range and maneuverability of the vehicle. However, their autonomy property directly affects the design of the control system. This requires advanced controllers and specific control schemes for given tasks.

Almost all AUVs are six degrees-of-freedom (DOF) systems, and various types of actuator configurations are available in the industry for the vehicles ranging from fully-actuated vehicles to underactuated ones. The vehicle of interest here falls into the class of underactuated AUVs. Any mechanical system having fewer actuators than its degrees of freedom is defined as an underactuated system. Some examples of underactuated systems include manipulators; (Arai et al., 1998), (Oriolo & Nakamura, 1991), (Yabuno et al., 2003), marine vehicles; (Reyhanoglu, 1997), (Pettersen & Egeland, 1996), space robots; (Tsiotras & Luo, 1997), and the examples given in (Fantoni & Lozano, 2002).

Controlling all of the DOF of underactuated mechanical systems is an arduous task compared to the fully actuated systems since the mathematical analysis of the system renders it difficult. Determining whether an underactuated system is controllable is one of these difficulties encountered. Control synthesis is also another challenge in this field and is still accepted as an open problem. The techniques used for fully actuated systems cannot be used directly for underactuated systems. However, there are some potential benefits over fully actuated systems depending on the efficiency of control and the task. In case of actuator failures, a fully actuated mechanical system falls into the class of underactuated systems and might still be controlled if a successful control scheme can be designed. Besides that, reduction of the weight and cost, and the increase of reliability can be considered as

advantages of underactuated systems. On the other hand, underactuation may take place by design as in helicopters, ships, underwater vehicles, satellites, hovercrafts, etc.
Control problem of underactuated systems has been generally studied as a control problem of a class of nonholonomic systems, although the relation between underactuated systems and nonholonomic systems has not been clear yet (Kolmanovsky & McClamroch, 1995). Nonholonomic systems are known as the mechanical systems of nonholonomic constraints which cannot be integrated to obtain the equations describing the position of the system. Control of nonholonomic systems poses a difficult problem requiring a special control approach depending on the nature of the mechanical system, and the modelling of nonholonomic systems as state equations is another difficulty (Sampei et al., 1999).
In spite of having six effective DOF, the vehicle has one controllable DOF since it has just one actuator (the propeller). The vehicle does not have any other control element save for the thrust provided by the propeller. The propeller produces the main thrust. Consequently, the reaction of the body to the load torque of the propeller produces a moment with respect to its rotational axis. Thus, the vehicle is considered underactuated because it has fewer actuators than the degrees of freedom of the system. The vehicle is a nonlinear system: all equations of motion of the system include coupled terms. Some equations of the motion of the system appear as second-order nonholonomic constraints, and they cannot be integrated to obtain position. Therefore, such underwater vehicle is pertained as a nonholonomic system.
In control of underactuated autonomous underwater vehicles (UUVs), optimal control approach has not been widely applied. Jeon et al. (Jeon et al., 2003) proposed an optimal linear quadratic controller for a 6-DOF underwater vehicle with four thrusters to distribute the thrust optimally. Additionally, some motion planning approaches were discussed in (Bøerhaug et al., 2006) and (Bullo & Lynch, 2001).
Fukushima (Fukushima, 2006) proposes a novel control method of solving optimal control problems including nonlinear systems. In this study, this control method is used. His method proposes a wide range of applicability and simplicity. It can be applied to both linear systems and nonlinear systems including the systems to be controlled in real-time. Although this method has similarities with the classical optimal control theory, it can be seen as a radical contribution in the control engineering field, rather than the extension of the existing optimal control theory.
Fukushima's method is fundamentally based on the employment of the energy generation, storage and dissipation of the controlled system. The total system energy stored in the system boundary is the sum of each energy. The criteria function consists of the control-performance, which is determined for a given task, the input energy, and the energy equation. First derivatives of the energy equation and the performance measures constitute a scalar function. The minimization of the scalar function yields the optimal control law. The necessary condition for the minimization is the Euler equation. The use of energy equation in the criteria function enables the optimal control law to have efficient dissipation characteristics. Obtaining the control law is a simple process and is not much mathematically involved. As one of the important properties of his control method, the control-performance can be of any form. There is no restriction in determining it, whereas the classical optimal control theory works well with performance measures of quadratic form.

2. Mathematical modelling of the UUV

The mathematical model of the UUV is derived in this section. The kinematics and dynamics of the UUV are studied assuming it as a rigid body. The kinematic equations, dynamic equations for the rigid body, and fluid-body dynamics are needed in the discussion of the UUV's motion. Combining those equations, the nonlinear model of the UUV as 6-DOF equations of motion is obtained. In this study, derivation of the mathematical model is not discussed in full detail. The reader is referred to (Meirovitch, 1970) and (Fossen, 1994) for a detailed treatment.

2.1 Description of the system

To obtain the mathematical model, the specifications and the assumptions which enable the formulation of kinematic and dynamic equations are needed. The simple model of the UUV holds the essential dynamical properties. A hull having a scalene ellipsoidal shape is shown in Fig.1. As illustrated in Fig.1, the UUV has a flat type of hull. For simulations, the UUV is assumed as neutrally buoyant and completely rigid. The fluid, which the vehicle interacts with, is considered as an ideal fluid (unbounded, irrotational, inviscid, and incompressible) and is chosen as sea water in simulations. Additionally, added mass related contributions in the equations of motion are neglected.

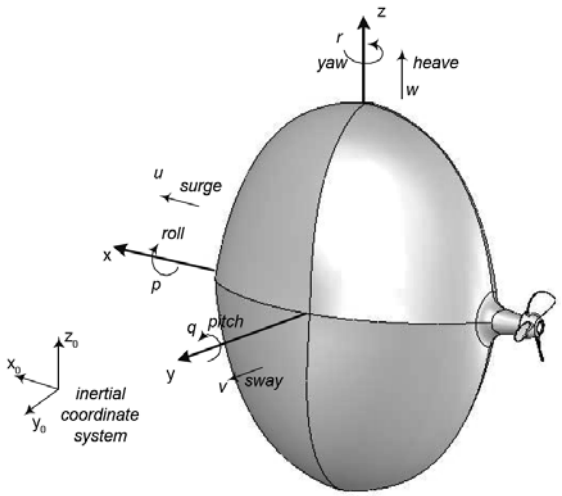


Fig. 1. Inertial and body-fixed coordinate systems and motion representations

2.2 Nonlinear model representation of the dynamic equations of motion

The motion of the UUV in space needs to be defined with respect to some certain coordinate frames. One of the coordinate frames can be chosen to be fixed to the vehicle and is called the body-fixed reference frame. The advantage of defining the motion of the UUV in terms of the linear and angular motion components about the orthogonal body axes leads to define the body-fixed frame, *xyz*. The body-fixed frame is chosen so as to coincide with the center of buoyancy (CB), which is the volumetric center of the fluid displaced. It implies that the CB vector is zero vector, $r_b = [x_b, y_b, z_b]^T = [0,0,0]^T$. Another orthogonal coordinate system is defined to describe the motion of the moving body-fixed frame relative to an inertial frame.

The earth-fixed reference frame, $x_0y_0z_0$, is assumed as fixed in Earth and accepted as inertial. These right-handed frames are shown in Fig.1.

The rotation of a rigid-body can be represented in many ways. The well-known and mostly used representation is Euler angles. This representation is practical, popular and has intuitive physical meaning. However, Euler angle parameterization causes some singularities (Ang & Tourassis, 1987) and inaccuracies in calculations, e.g. discontinuous changes may occur in the attitude when the rotation is changed incrementally. A singularity-free and well-suited quaternion parameterization is preferred for accurate calculations.

The transformation between the body-fixed frame and the earth-fixed frame is given by (Fossen, 1994):

$$\dot{\eta}_E = E(\eta_E)\nu\,, \tag{1}$$

where E is the transformation matrix and $\eta_E=[x,y,z,\varepsilon_1, \varepsilon_2, \varepsilon_3,\eta]^T$. Here, $e=[\varepsilon_1, \varepsilon_2, \varepsilon_3,\eta]$ is a unit quaternion vector. The unit quaternion vector represents the rotation with respect to an axis of rotation, $\varepsilon=[\varepsilon_1, \varepsilon_2, \varepsilon_3,\eta]^T$, and a rotation of angle, η, about that axis. The linear and angular velocities of the vehicle are described as $\nu = [u,v,w,p,q,r]^T$.

The dynamic equations of motion for the UUV can be written as follows (Fossen, 1994):

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = \tau\,, \tag{2}$$

where $\dot{\nu}$ is the time derivative of the velocity vector, and τ is propulsion forces and moments vector. The simplified inertia matrix, M, is written as

$$M = \mathrm{diag}\left\{m,m,m,I_X,I_Y,I_Z\right\}, \tag{3}$$

where m is the mass of the UUV, and I_x, I_y, and I_z are the moments of inertia. Coriolis and centripetal matrix, $C(\nu)$, is given as

$$C(\nu) = \begin{bmatrix} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ 0 & mw & -mv & 0 & I_Z r & -I_Y q \\ -mw & 0 & mu & -I_Z r & 0 & I_X p \\ mv & -mu & 0 & I_Y q & -I_X p & 0 \end{bmatrix}. \tag{4}$$

D(v) is the damping matrix which includes only terms of quadratic drag:

$$D(\nu) = -\mathrm{diag}\left\{X_{u|u|}|u|, Y_{v|v|}|v|, Z_{w|w|}|w|, K_{p|p|}|p|, M_{q|q|}|q|, N_{r|r|}|r|\right\}, \tag{5}$$

where, $-X_{uu}$ is the coefficient of the drag which the vehicle experience due to the motion along the x-axis, and $-N_{rr}$ is the coefficient of the hydrodynamic torque due to the rotational motion of the vehicle with respect to the z-axis. Other coefficients can be described similarly.

The effect of environmental disturbances such as waves and ocean currents are neglected in this study. As a result, following six differential equations describing the 6-DOF equations of motion in surge, sway, heave, roll, pitch, and yaw, respectively, are obtained:

$$m\left(\dot{u} + qw - rv\right) - X_{uu}\, u|u| = F_x$$

$$m\left(\dot{v} + ru - pw\right) - Y_{vv}\, v|v| = 0$$

$$m\left(\dot{w} + pv - qu\right) - Z_{ww}\, w|w| = 0$$

$$I_x \dot{p} + -\left(I_z - I_y\right) qr - K_{pp}\, p|p| = k_1 F_x$$

$$I_y \dot{q} + \left(I_x - I_z\right) rp - M_{qq}\, q|q| = 0$$

$$I_x \dot{p} + \left(I_z - I_y\right) qr - K_{pp}\, p|p| = k_1 F_x . \tag{6}$$

The main thrust is denoted by F_x, thus, the propulsion forces and moments vector can be written as $\tau=[F_x,0,0,k_1F_x,0,0]^T$, where k_1 is a coefficient relating the ratio of the thrust to the rolling moment of the body. The load torque of the propeller causes the body to react with an equal torque and to rotate in opposite direction. The reaction torque is another control input to the system, but, obviously, it is dependent on the thrust. Since the mechanical model does not have control fins, (6) does not include any input for control surface forces and moments. In (6), except for the equations in surge and roll, four equations can be identified as second-order nonholonomic constraints which expose the non-integrable velocity relationships. These constraints imply that possible displacements of the body in each direction are not independent, but are mutually connected.

3. Control method

Fukushima proposed a control method of solving optimal control problems including nonlinear systems. Although, in Fukushima's method, there are similarities with the classical optimal control theory, it introduces a new optimal control approach which is fundamentally based on the employment of the energy generation, storage and dissipation of the controlled system. The total system energy stored in the system boundary is the sum of each energy. The rate of change of the instantaneous energy yields the net power flow of the dynamical system. Hence, the general power balance equation for a controlled system can be represented as follows:

$$P = u^T \dot{q} + v^T \dot{q} - \ddot{q}^T M(q)\dot{q} - d^T(q,\dot{q})\dot{q} - e^T(q,z)\dot{q} = 0, \tag{7}$$

where u is the input force vector, v is the input disturbance force vector, $M(q) \in R^{n^x n}$ is the symmetric positive-definite inertia matrix with n, the number of the DOF of the controlled system, d is combination of the damping force vector, and Coriolis and centrifugal force

vector, e is the potential force vector, z is the input disturbance displacement vector, and q is the generalized coordinates vector. The power equation of the system has dynamic characteristics of the controlled system. It is unique to the system and has an important role in the design of the optimal control system.

In optimal control theory, it is aimed to obtain a control law which satisfies the given constraints and extremizes the performance measure. A performance measure is mostly a combination of some scalar functions. The functional below is the performance measure used in Fukushima's control method:

$$J = \int (g(q, \dot{q}, \ddot{q}) + r' u^T \dot{q}) dt \,, \tag{8}$$

where g is the performance criterion (control performance) to be selected for the given control problem. Combination of these two terms is known as the performance measure for a general optimal control system. In (8), g might represent the linear combination of more than one performance criterion. The use of multiple performance criteria depends on the selection of performance measures for the control objective. In (8), the integrand of the last term represents the power delivered to actuators; here u^T is the transpose of the input force vector, and r' is a weighting factor.

According to the fundamental theorem of the calculus of variations (Elsgolc, 1961), the necessary condition for minimizing the performance measure is that the first variation of the functional must be zero. In Fukushima's method, the scalar function, L, is composed of the power equation (7) and the differential of the performance measure (8):

$$L = \kappa(u^T \dot{q} + v^T \dot{q} - \ddot{q}^T M(q)\dot{q} - d^T(q, \dot{q})\dot{q} - e^T(q, z)\dot{q}) + g(q, \dot{q}, \ddot{q}) + r' u^T \dot{q} \,, \tag{9}$$

where κ is an undecided constant. The included power balance equation is zero because it satisfies the energy conservation law. As the performance measure is minimized by means of the calculus of variations, L can be also minimized. The Euler equation is a necessary condition for minimization. After applying the Euler equation, the optimal control law is obtained.

4. Optimal control of the UUV

In this section, application of the introduced control method is discussed. Before formulating the optimal control problem, the constraints, performance criterion and the criteria function is given. The application of the method to the control problem yields the optimal control law.

4.1 Constraints

In this section, the constraints on the state and control values are defined. Let t_0 is the initial time and t_f is the final time, then, the state constraints are given as:

$$\begin{bmatrix} x_0(t_0) \\ y_0(t_0) \\ z_0(t_0) \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}; \qquad \begin{bmatrix} x_0(t_f) \\ y_0(t_f) \\ z_0(t_f) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{10}$$

where x_i, y_i, and z_i are initial positions. There are no constraints imposed on attitude and velocities of the UUV. Control constraints are imposed on the physical systems to be controlled due to the limitations of actuators. Force or torque inputs are bounded by some upper limit. In control of the UUV, it is assumed that there is no constraint on control, since the aim of this study is to show the possibility of control of such a challenging mechanical system. However, the simulations are also done for considering the constraints on the thrust:

$$-T_c \leq F_x \leq T_c, \tag{11}$$

where T_c is the thrust value.

4.2 Performance criterion

As a performance criterion, the minimum distance between two points has chosen to transfer the system from a point to another. The optimal control will try to minimize this measure. For this performance, only position of the UUV is of interest. When missions of underwater vehicles are considered, mostly, the positioning of the underwater vehicle is carried out. Thus, the performance criterion describing the minimum distance between two points in 3-D space can be written as a terminal-error function:

$$J_c = (x_f(t) - x_d)^2 + (y_f(t) - y_d)^2 + (z_f(t) - z_d)^2. \tag{12}$$

J_c represents the deviation of the actual path of the system from its desired path. The components of the final position vector, x_d, y_d, and z_d, are chosen as constant. For the calculation of the functions, $x_f(t)$, $y_f(t)$, and $z_f(t)$, which give the actual position of the vehicle, the relation (1) is used.

4.3 Criteria function

As was explained above, the criteria function consists of the control-performance (performance criterion), the performance measure describing the control effort, and the energy equation of the controlled system is given by

$$J = J_c + \int r' u^T v dt + \int P dt, \tag{13}$$

where the second term on the right-hand side of the equation is the power delivered to the actuator, and r' is the weighting factor which is determined as equal to the number of actuators. This control effort term can be denoted as J_e and written as:

$$J_e = r' \int F_x u dt \tag{14}$$

The last term in (13) is the energy equation. The total power equation, P, is the sum of all power flows in the system:

$$P = P_{in} + P_{generated} - (P_{dissipated} + P_{stored} + P_{out}) = 0. \tag{15}$$

Since each equation corresponding to each DOF in the equations of motion, (6), represents a force or moment equality, each equation can be also formulated so as to represent a power

equality. If each term of each equation in (6) is multiplied by the corresponding velocity, the power equalities are obtained. The sum of all equations yields the total power equation:

$$\begin{aligned} & F_X (u + k_1 p) - m(\dot{u}u + \dot{v}v + \dot{w}w) - I_X \dot{p}p - I_Y \dot{q}q - I_Z \dot{r}r \\ & + X_{uu} u^2 \mid u \mid + Y_{vv} v^2 \mid v \mid + Z_{ww} w^2 \mid w \mid + K_{pp} p^2 \mid p \mid + M_{qq} q^2 \mid q \mid + N_{rr} r^2 \mid r \mid = 0 \end{aligned} \quad . \tag{16}$$

In the total power equation, the remaining terms after summation are the input, stored, and dissipated power terms. Among these terms, the dissipation terms are of great importance since they appear in the control law and have contribution on the stability of the system.

4.4 Optimal control problem

Consider finding an admissible control u_{opt} which causes the system

$$\dot{x} = f(x, \tau) , \tag{17}$$

to follow an admissible trajectory, x^*, that minimizes the criteria function

$$J = J_C + J_e + \int \kappa P \, dt , \tag{18}$$

so that u_{opt} is called the optimal control. In (18), the term κ is a weighting factor which is determined in search of the minimum J.
Let criteria function be formulated as follows:

$$\begin{aligned} J &= \int (r' F_X u + P + \frac{dJ_C}{dt}) dt \\ J &= \int L \, dt \end{aligned} , \tag{19}$$

where the scalar function L is written as

$$L = 2(x_f \dot{x}_f + y_f \dot{y}_f + z_f \dot{z}_f) + F_X r' u + \kappa P . \tag{20}$$

In the calculus of variations, it is well known that a necessary condition for x^* to be an optimal of the functional given by (19) is (Naidu, 2003)

$$\frac{\partial L}{\partial x} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = 0 . \tag{21}$$

This equation is called Euler equation. Application of Euler equation to (19) yields the optimal control law:

$$u_{opt} = \rho_1 \frac{3\kappa X_{uu} u \mid u \mid}{\kappa + r'} + \rho_2 \frac{3K_{pp} p \mid p \mid}{k_1} - \rho_1 \frac{2T_e}{\kappa + r'} \tag{22}$$

with

$$T_e = x_f (1 - 2(\varepsilon_2^2 + \varepsilon_3^2)) + y_f (\varepsilon_1 \varepsilon_2 + \varepsilon_3 \eta) + z_f (\varepsilon_1 \varepsilon_3 - \varepsilon_2 \eta) , \tag{23}$$

where ρ_1 and ρ_2 are weighting factors. These weighting factors and κ are computed by carrying out a multivariable optimization for the minimization of the terminal error function (12). Choice of the best control history for different values of these weighting factors leads to the determination of their values. When control constraint is not imposed on the system, it is obtained more than one admissible control history for different values of weighting factors. If control constraint is imposed on the system, there might still be admissible control histories.

In the optimal control law, (22), two points are of importance: the dissipation terms and the control-performance related terms. The terms coming from the power equation are the dissipation terms: $3X_{uu}u|u|$ and $3K_{pp}p|p|$. There is not any other term reflecting system dynamics in the control law. The input to the UUV is effective on its two DOF. The surge and roll motions are controlled by the thrust. Therefore, the input related dissipation terms appear in (22). The second important term is T_e and it is dominantly effective than dissipation terms in positioning the vehicle.

5. Simulation results

Using the derived control law (22), numerical simulations of the described ellipsoidal UUV were performed for two cases via *Matlab/Simulink™* software tool. In the first case, the control input is not constrained, whereas the second case shows the simulations with the constrained input. In the second case, the initial position has been chosen different than the first case. For both cases, initial velocities of the underwater vehicle are given as: $[u(0),v(0),w(0),p(0),q(0),r(0)]^T=[0,0,0,0,0,1]^T$. The values of the parameters used in the simulations are given in Table 1.

Parameter	*Value*	*Unit*	*Description*
a	0.2	m	Equatorial radius(X)
b	0.15	m	Equatorial radius(Y)
c	0.25	m	Polar radius(Z)
m	32	kg	Vehicle mass
ρ	1030	kg/m^3	Seawater density
I_x	0.55	kgm^2	Moment of inertia
I_y	0.66	kgm^2	Moment of inertia
I_z	0.4	kgm^2	Moment of inertia
ρ_1	- 1.15	n/a	Weighting factor
ρ_2	- 0.007	n/a	Weighting factor
κ	0.0065	n/a	Weighting factor
k_1	0.025	m	Weighting factor
r'	1	n/a	Weighting factor

Table 1. Simulation parameters

5.1 Case-1

In this case, the initial position and attitude of the vehicle with respect to the earth-fixed frame are $[x_0(0),y_0(0),z_0(0),\varepsilon_1(0),\varepsilon_2(0),\varepsilon_3(0),\eta(0)]^T= [50,50,50,0,0,0,1]^T$ at time $t_0=0$. The target point is the origin, $[x_d,y_d,z_d]^T=[0,0,0]^T$. There is no constraint on the control input.

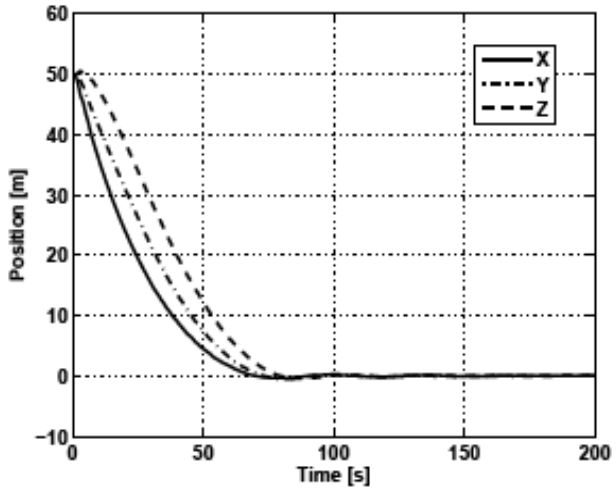


Fig. 1. Time evolution of the position of the UUV

As shown in Fig.1, the UUV achieved to reach the target point at about *t=80*. After reaching the target, the vehicle could keep its position very close to the origin as the variation of linear and angular velocities indicate in Fig.3. The control law, Fig.2, has oscillating and damping characteristics, and the frequency of the control signal is observed as *f=0.5Hz* at most.

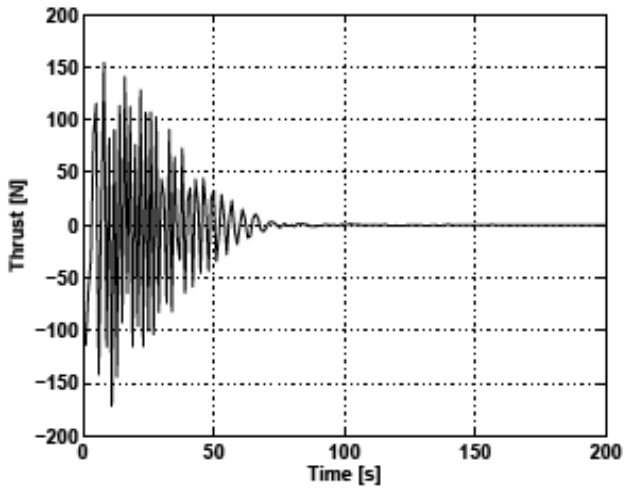


Fig. 2. Time evolution of the applied input

5.2 Case-2

In this case, the initial position and attitude of the vehicle with respect to the earth-fixed frame are $[x_0(0), y_0(0), z_0(0), \varepsilon_1(0), \varepsilon_2(0), \varepsilon_3(0), \eta(0)]^T = [-40, 25, -15, 0, 0, 0, 1]^T$ at time $t_0=0$. The target point is the origin, $[x_d, y_d, z_d]^T=[0,0,0]^T$. The following thrust constraint (in Newtons) has been imposed on the control input:

$$-25 \leq F_X \leq 25 ,$$

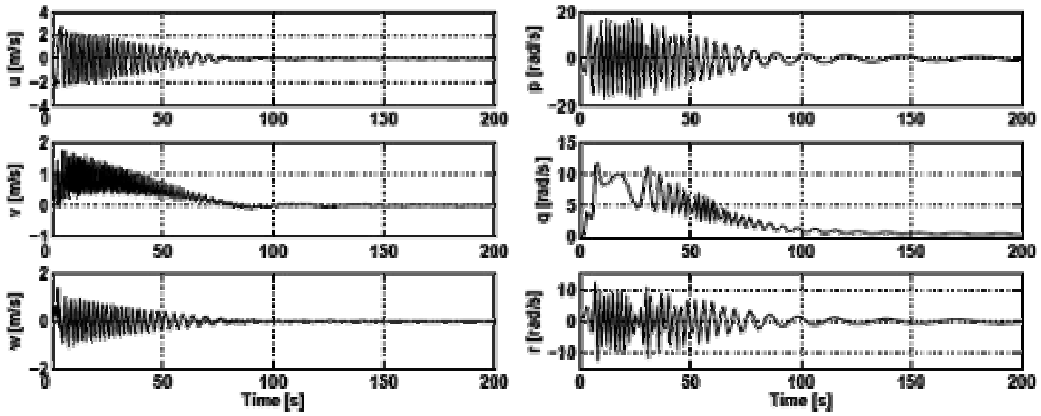


Fig. 3. Time evolution of the linear and angular velocities

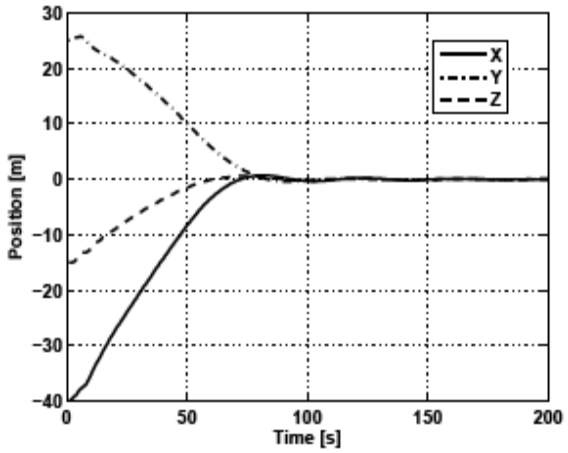


Fig. 4. Time evolution of the position of the UUV

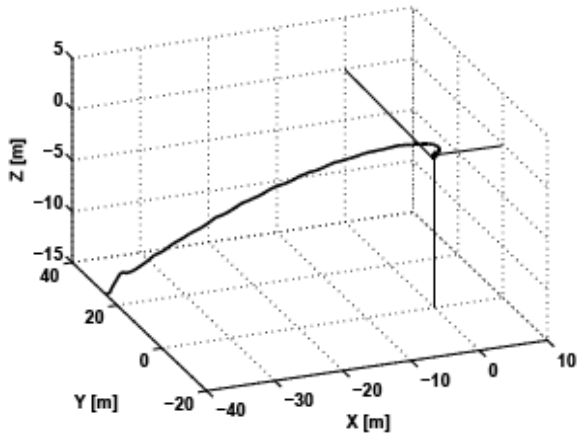


Fig. 5. 3-D flight path of the UUV

In this case, the only difference between applying an unconstrained input and constrained input is that the operation time is longer in the latter case. The constrained control, Fig.6, has a similar characteristic with that of the previous case.

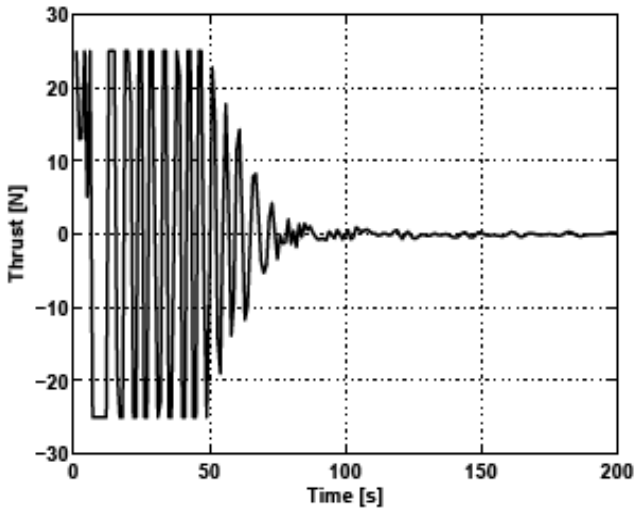


Fig. 6. Time evolution of the applied input

6. Conclusion

In this study, position control of the UUV as the application of a novel optimal control was discussed. The dynamics and control of the 6-DOF UUV are presented and the new optimal control method was introduced. To the best knowledge of the authors, there is no 6-DOF underwater vehicle that can be controlled with one actuator in literature. Based on this control approach, the criteria function determined included a terminal error function. The terminal error function plays an important role in control of the UUV. After the minimization of the criteria function, the terms coming from the terminal error function appeared in the control law. After simulations, it was observed that these terms are crucial in the maneuver of the vehicle. It is understood that they dominantly, with respect to the other terms, determine the value and direction of the force applied to the UUV.

It has been proved that, the control of the UUV necessitates the initial conditions as either pitch velocity or yaw velocity at time $t_0=0$. The roll velocity is produced by the rolling moment. Gyroscopic moments and centrifugal forces occur when the pitch or yaw velocity has an initial value. Existence of these forces and moments make the control possible. Even though the use of only one actuator seems to be a big challenge for a 6-DOF underwater vehicle, the UUV has succeeded to reach the target point by application of the energy-based control method.

At present, attitude control of the UUV has not been achieved. Research for controlling the attitude variables is ongoing. Even the vicinity of the target state is attained, the vehicle may still have small velocities. The body needs forces in each direction for a necessary maneuver. Since the nonholonomic constraints do not allow every possible motion, motion to minimize the position errors needs other necessary forces to be produced by only F_x. Therefore, the

body moves undesired directions recurrently. If attitude control would be achieved, this hovering motion is thought to be prevented.

7. References

Ang, J. M. H. & Tourassis, V. D. (1987). Singularities of euler and roll-pitch-yaw representations, IEEE Transactions on Aerospace and Electronic Systems, vol. AES-23, no. 3, pp. 317–324

Arai, H.; Tanie, K. & Shiroma, N. (1998). Nonholonomic control of a three-DOF planar underactuated manipulator, IEEE Transactions on Robotics and Automation, vol. 14, no. 5, pp. 681–695

Bøerhaug, E.; Pettersen, K.Y. & Pavlov, A. (2006). An optimal guidance scheme for cross-track control of underactuated underwater vehicles, *14th Mediterranean Conference on Control and Automation*, pp.1-5, June 2006

Bullo F. & Lynch, K.M. (2001). Kinematic controllability for decoupled trajectory planning in underactuated mechanical systems, *IEEE Transactions on Robotics and Automation*, vol.17, no.4, pp.402-412

Elsgolc, L. E. (1961). *Calculus of Variations*, Pergamon Press

Fantoni I. & Lozano, R. (2002). *Non-lienar Control for Underactuated Mechanical Systems*, Springer-Verlag, London

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley & Sons, New York

Fukushima, N. (2006). Optimal control of mechanical system based on energy equation, *Trans. of JSME(C)*, vol. 72, no. 722, pp.3106–3114

Jeon, B.-H.; Lee, P.-M.; Li, J.-H.; Hong, S-W; Kim, Y.-G. & Lee J. (2003). Multivariable optimal control of an autonomous underwater vehicle for steering and diving control in variable speed, *Proceedings of OCEANS 2003*, vol.5, pp.2659-2664

Kolmanovsky, I. & McClamroch, N. H. (1995). Developments in nonholonomic control problems, *IEEE Control Systems*, vol. 15, pp. 20–36

Meirovitch, L. (1970). *Methods of Analytical Dynamics*, McGraw-Hill

Naidu, D. S. (2003). *Optimal Control Systems*, CRC Press, London

Oriolo, G. & Nakamura, Y. (1991). Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators, *Proceedings of IEEE Conf. Decision and Control*, pp. 2398-2403, Brighton, U.K.

Pettersen K. & O. Egeland (1996). Position and attitude control of an underactuated autonomous underwater vehicle, *Proceedings of the 35th Conference on Decision and Control*, pp. 987–991, Kobe, Japan

Reyhanoglu, M. (1997). Exponential stabilization of an underactuated autonomous surface vessel, *Automatica*, vol. 33, no. 12, p. 2249-2254

Sampei, M.; Kiyota, H. & Ishikawa, M. (1999). Control strategies for mechanical systems with various constraints-control of non-holonomic systems, *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 3, pp. 158–165

Tsiotras, P. & Luo, J. (1997). Reduced-effort control laws for underactuated rigid spacecraft, *Journal of Guidance, Control, and Dynamics*, vol. 20, pp. 1089–1095

UNECE/IFR (2005). *World robotics survey 2005, press release*, ECE/STAT/05/P03, Geneva, 1 October 2005.

Yabuno, H.; Matsuda, T. & Aoshima, N. (2003). Motion control of an underactuated manipulator without feedback control, *Proceedings of 2003 IEEE Conference on Control Applications*, vol. 1, pp. 700–705

3

Navigating Autonomous Underwater Vehicles

Brian Bingham
Franklin W. Olin College of Engineering
U.S.A

Navigation is the process of directing the movements of a ship or aircraft form one point to another. Both art and science are involved in conducting a ship safely to its destination.

(Dunlap, 1975)

1. Introduction

Autonomous Underwater Vehicles (AUVs) are powerful tools for exploring, investigating and managing our ocean resources. As the capabilities of these platforms continue to expand and they continue to mature as operational assets, navigation remains a fundamental technological component.

This chapter presents a road map for the vehicle designer to aid in integrating the latest navigation methods into new platforms for science, industry and military platforms. Along the way, we point to emerging needs where new research can lead directly to an expansion of the operational abilities of these powerful tools. To accomplish this we start by describing the problem, explaining the needs of vehicle users and the challenges of autonomous localization. Next we explain the state of practice, how operational assets currently solve this difficult problem. To expand this explanation we present new research targeted at helping AUV builders to make the complex tradeoffs in creating a platform with the appropriate navigation solution. We conclude with an overview of the latest research and how these advances might soon become available for AUV operations in new environments such as the littoral zone, at the poles and under-ice. Throughout this chapter we attempt to reach across the disciplinary boundaries that separate the researcher from the operator.

2. Motivation

2.1 The challenge of autonomous underwater navigation

Navigating an AUV presents unique challenges to the researcher and the practitioner. One way to understand the particularities of this challenge is to consider two important facets of AUV operations: the marine environment and desired results.

The ocean environment presents both challenges and opportunities for autonomous navigation. The challenges are well documented: seawater is opaque to electromagnetic signals making Global Positioning System (GPS) solutions infeasible; acoustic communication

is limited in bandwidth, scale and reliability (Catipovic, 1990) and the ocean environment is observationally limited and ever-changing.
On the other hand the deep-sea environment can be an ideal place for autonomous vehicle operations. The unstructured environment can be structured by the addition of acoustic transponders moored to the seafloor or through close communication with a surface ship. Either method provides an absolute position reference which decreases the demands on real-time perception and decision making. Also, deep-water can be one of the most forgiving acoustic environments because of the homogeneous and stable sounds speed structure and low ambient noise.
The opportunity for novel observation counterbalances these operational difficulties. We have better maps of Mars, Venus and the Moon that we have of the Earth's ocean, creating a great potential to advance our observational capability through technology.

2.2 Creating new data products

Typically a gap between the needs of the AUV user and the capabilities of the navigation solution. The user is often not directly interested in the navigation, but instead is focused on producing a *data product*, an gestalt representation of the underwater environment. The vehicle designer should incorporate the right navigation instruments and the right data processing to provide a navigation solution appropriate for the desired data product. This perspective, having the requirements of the data product drive the design decisions, leads to closing the gap illustrated in Fig. 2.

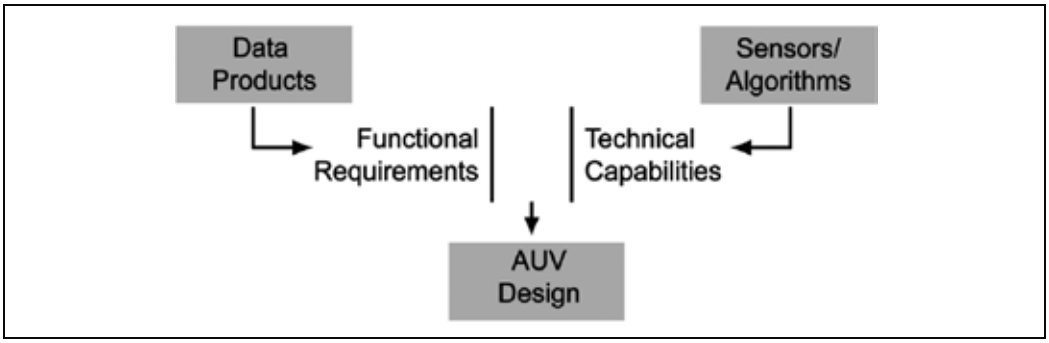


Fig. 1. Illustration of how vehicle design decisions are driven by the needs of the application (the desired data product) and the capabilities of the navigation sensors and algorithms.

It is only a slight over simplification to consider the resolution of any observation to be directly proportional to the navigation precision. Fig. 2 shows a common situation to illustrate this notion. In this case the data products are a photomosaic and a small-scale bathymetry map, both shown in the figure. The remotely operated vehicle (ROV) JASON is shown as it surveys the seafloor. Navigation allows all the measurements (e.g., sonar bathymetry) and observations (e.g., optical images) to be placed in a common coordinate system. How well we can resolve two disparate data sources, i.e., the resolution of our data product, depends on the uncertainty in our navigation. Summarized another way, the spatial size of each "pixel" in our final image is fundametnally limited to the uncertainty in our navigation solution.

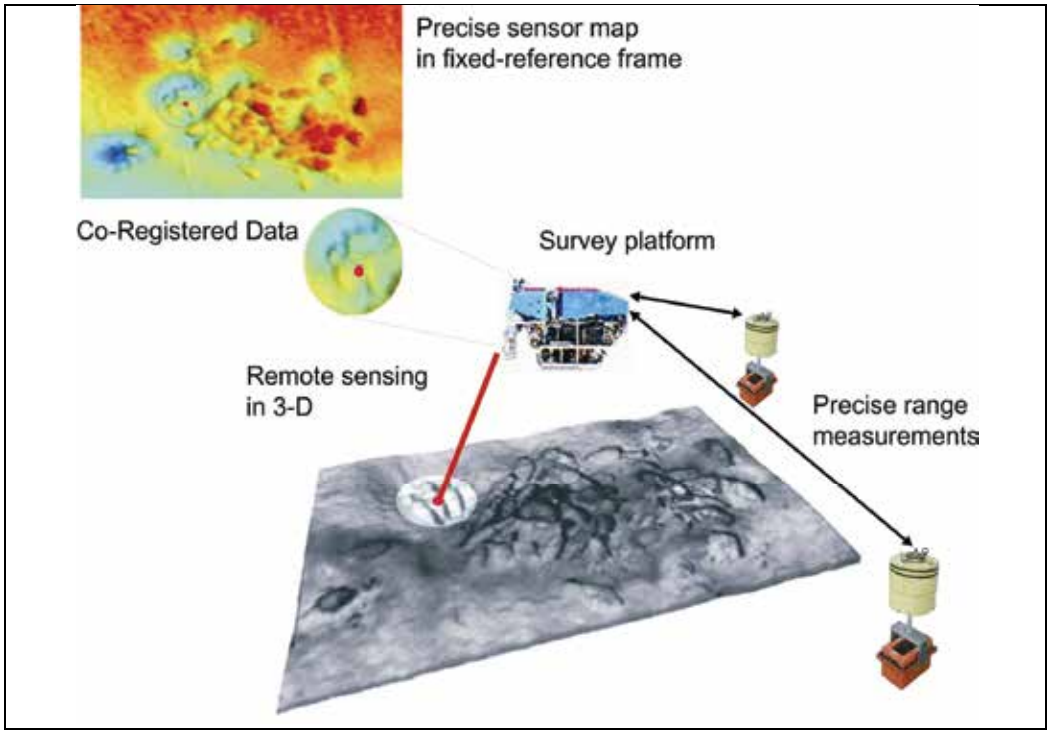


Fig. 2. Illustration of the concept of co-registered data. The ROV JASON is shown performing a survey collecting optical images and bathymetry data. Range-based navigation provides a common coordinate system. Component images are courtesy of the Deep Submergence Lab (DSL) at the Woods Hole Oceanographic Institution.

3. State of practice

AUV operations require a reliable navigation solution. Methods currently in operation on autonomous platforms are simple and robust. These real-world solutions typically make use of just a few key sensors:

- GPS receivers to measure position at the surface
- Long baseline transponders to measure the distance from the AUV to transponders in known locations.
- Doppler velocity logs to measure velocity relative to the bottom, supported by attitude and heading measurements

These sensors are dedicated navigation sensors, distinct from the remote sensing payload sensors which collect measurements which are not processed for real-time perception. These relatively simple sensing modalities, configured and combined in a variety of interesting ways, have proven to provide a variety of solutions that are robust to the complexities of the ocean environment.

3.1 An example

It is informative to consider a particular example. This example, like the data shown in Fig. 2, is taken from work with the JASON ROV system from the Deep Submergence Lab at

Woods Hole Oceanographic. The ROV is instrumented with a combination that has become standard in AUV and ROV applications: absolute positioning using LBL transponders and seafloor odometry from a DVL and heading reference.

To understand the tradeoffs in designing an appropriate navigation system it is useful to contrast modalities that exhibit unbounded error growth with those that have bounded error. Fig. 3 illustrates this contrast. The dead-reckoning solution provided by the DVL alone is shown to drift over time; the error growth is unbounded. In Fig. 3 the DVL track begins at the origin (shown in the figure as a large "X") and then diverges from the absolute reference provided by the LBL reference. In what follows we show how quantitative models of this error accumulation can be used to improve design and operation.

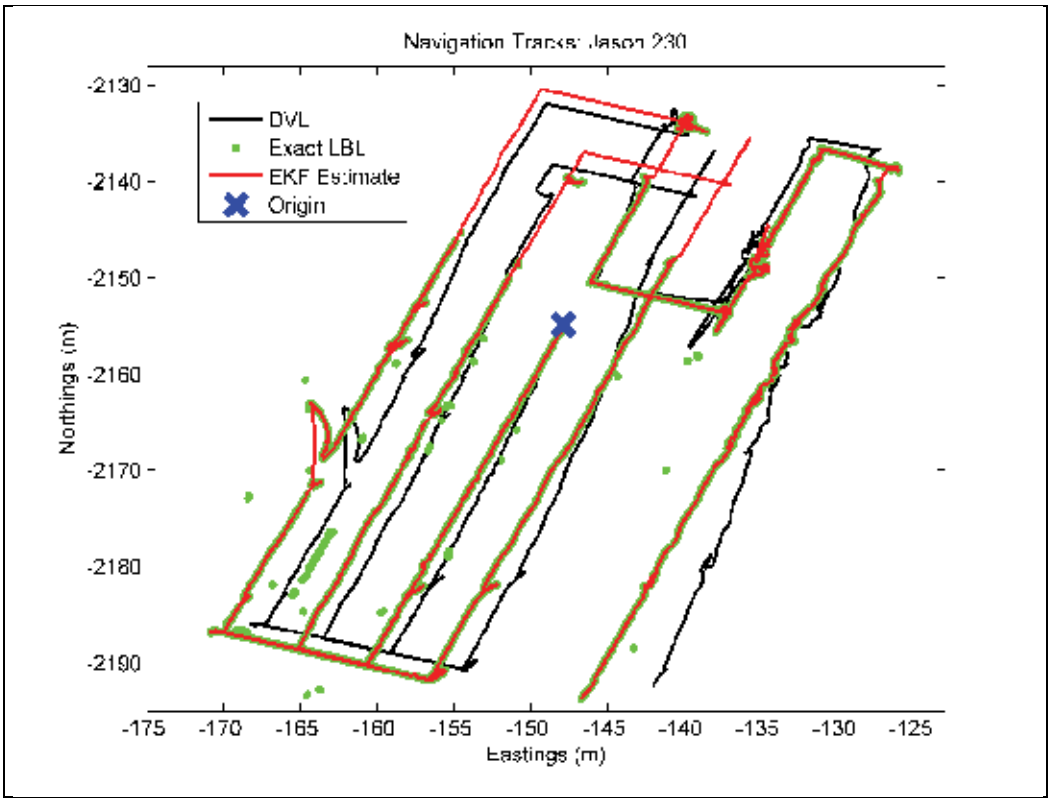


Fig. 3. Three navigation tracks from the ROV Jason, lowering #230. The "DVL" track shows the dead-reckoning resulting from the DVL odometry alone. The "Exact LBL" track shows the standalone LBL solution. The "EKF Estimate" track shows the combination of both the DVL and LBL information using an extended Kalman filter framework. All tracks are started at the "Origin". The tracklines were executed over 3.5 hours at an average depth of 2,265 m.

The LBL position solution complements the DVL dead-reckoning. Returning to Fig. 3 we see that the Exact LBL provides a solution with bounded uncertainty, but with a high degree of random errors or noise. We can see outliers (shown by widely spaced data points) and zones where no LBL is returns are received (eg., the Exact LBL track dissappears in the northwest corner of the figure). A particularly insidious form of error is the consistent, but

off-set position solutions shown in the southwest section of the survey. This type of error can be difficult to filter autonomously.
Finally, to illustrate the possibility of leveraging the complementary nature of the two navigation tracks, we show the results of an extended Kalman filter (EKF) estimator. This track uses absolute positioning from the LBL source to constrain the unbounded uncertainty in the DVL dead-reckoning. By simultaneously using both sources of information, the EKF solution combines the strengths of both methods. This example highlights the contrasts between the unbounded uncertainty of DVL dead-reckoning, the bounded uncertainty of LBL positioning and the utility of combining these two solutions.

3.2 Long Baseline (LBL) positioning

Long baseline (LBL) positioning is a standard in underwater navigation. First used in the 1960's and 1970's (Hunt, Marquet, Moller, Peal, Smith, & Spindel, 1974), the foundational idea of using acoustic transponders moored to the seafloor has been used to fix the position of a wide spectrum underwater assets: submersibles, towed instrumentation, ROVs and AUVs. Fig. 4 illustrates the basic LBL method for use with an AUV. For each navigation cycle the vehicle measures the two-way time-of-flight for an acoustic signal sent round trip between the platform and fixed transponders on the seafloor. Position is determined by multilateration, typically implemented as a non-linear least-squares solution to the spherical positioning equations.
Due to the particular challenges and constraints of working in marine environments, a large variety of range-based positioning solutions have been put into practice. The ability to precisely measure the range between two acoustic nodes is the foundation of any such solution. For example, short baseline (SBL) techniques are equivalent to the LBL positioning except that the transponders are in closer proximity, often mounted to the surface ship or platform (Milne, 1983) (Smith & Kronen, 1997). Wired configurations are used in small environments and allow one-way range measurement (Bingham, Mindell, Wilcox, & Bowen, 2006). Such solutions can be particularly useful for confined environments such as small test tank (Kinsey, Smallwood, & Whitcomb, 2003).
There are many implementations of the basic LBL positioning method. Commerical systems are available to provide support for scientific, military and industry application. Typical systems operate at frequencies near 10 kHz with maximum ranges of 5-10 km and range resolution between 0.5 and 3 m[1]. Specific purpose systems are also available for small-scale high-resoution positioning[2] or even subsea geodetics.
Fig. 5 is a conceptual sketch of the method of spherical positioning which can be generalized with a stochastic measurement model. Each spherical positioning solution is based on observing individual range values (z_{r_i}) between known fixed beacon locations ($\boldsymbol{x}_{b_i}$) and an unknown mobile host position ($\boldsymbol{x}_h$) where the individual range measurements is indexed by i.

$$z_{r_i} = \left|\boldsymbol{x}_h - \boldsymbol{x}_{b_i}\right| + \omega_{r_i} \tag{1}$$

We consider the additive noise in each measurement (ω_{r_i}) as an independent, zero-mean, Gaussian variable with variance σ_r^2.

[1] Examples include solutions from Teledyne Benthos, Sonardyne International Ltd. and LinkQuest Inc.
[2] Examples include solutions from Desert Star Systems or Marine Sonics Technology, Ltd.

$$\omega_{r_i} \sim \mathcal{N}(0, \sigma_r^2) \quad (2)$$

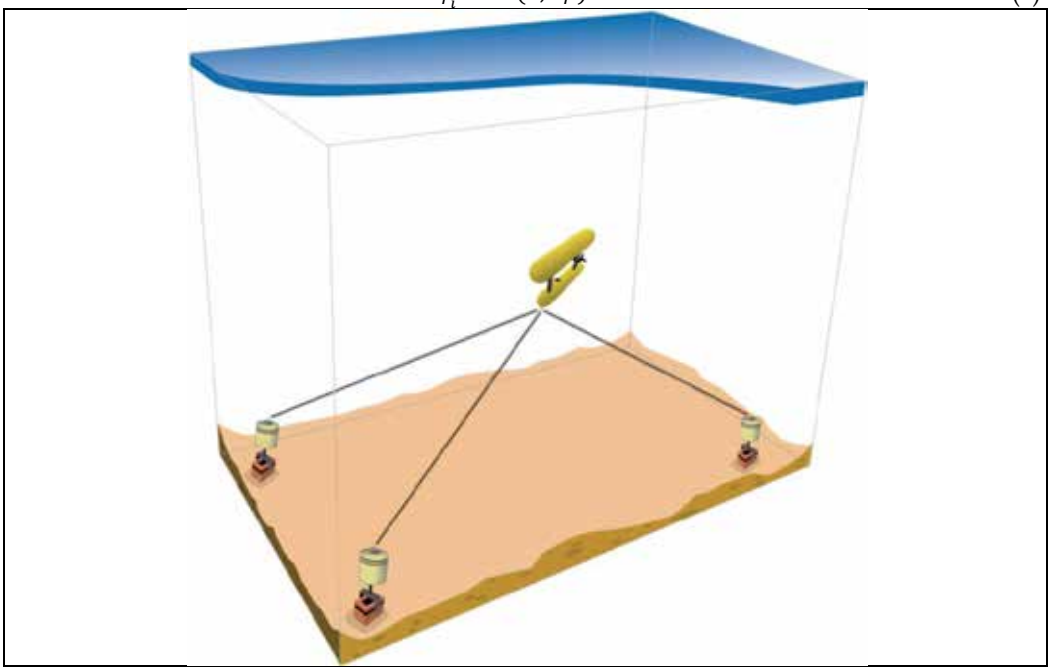


Fig. 4. Illustration of long baseline (LBL) positioning of an AUV in an instrumented environment. Three transponders are shown moored to the seafloor. Three time-of-flight range observations are represented by dashed lines between the seafloor transponders and the mobile host, in this case an autonomous underwater vehicle.

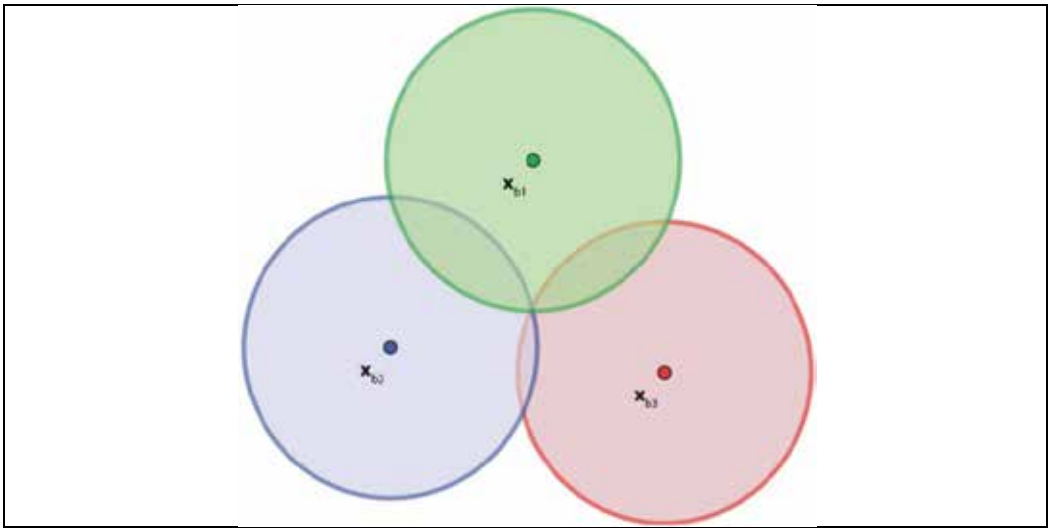


Fig. 5. Illustration of a standalone spherical positioning solution, shown in two dimensions. Each of the three transponders is represented by a mark at the center of the three circles

(x_{b_i}). By measuring a range from each transponder we know the radius of each circle. With three ranges the position is estimated by the intersection of the three circles.

3.3 Doppler Velocity Log (DVL) dead-reckoning

A Doppler velocity log (DVL), integrated with a precise heading reference, is another standard instrument for underwater robotics. As a standalone solution, DVL navigation provides a dead-reckoning estimate of position based on discrete measurements of velocity over the seafloor. To produce this dead-reckoning estimate in local coordinates sequential DVL measurements are related to a common coordinate system. Because the raw measurements are made relative to the sensor, the attitude (heading, pitch and roll) of the sensor relative to the common coordinate system must be measured. Once compensated for attitude, the velocity measurements are accumulated to estimate position.

The position uncertainty for standalone DVL dead-reckoning grows with both time and distance. Fig. 6 illustrates a simple example of this error growth based on a vehicle moving at a constant speed along the x-axis. Velocity uncertainty causes uniform error growth in both directions while heading uncertainty dominates the error growth in the across track direction. To further quantify the dynamics of uncertainty in such a situation we propose an observation model compatible with the LBL uncertainty model presented above.

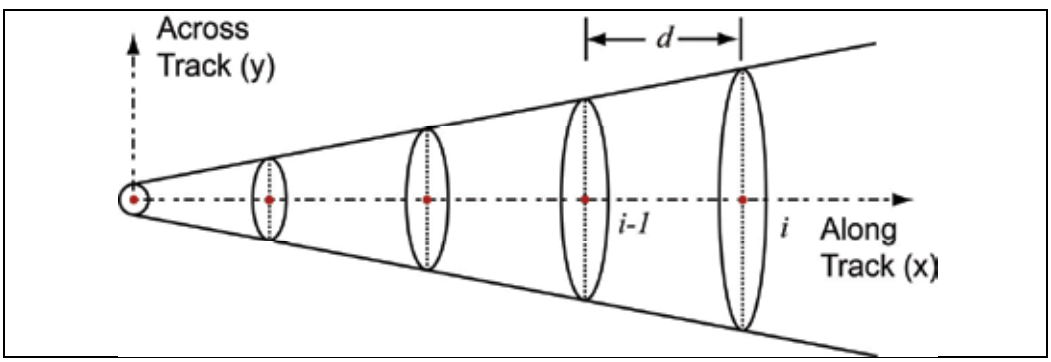


Fig. 6. Illustration of odometry uncertainty dynamics. The ellipses illustrate the 1-σ uncertainty in the along track (x) and across track (y) directions. Five discrete vehicle positions are shown, indexed by i. The distance between consecutive positions is indicated by d.

The DVL instrument provides independent measurements of velocity (z_{v_k}) in each of three dimensions (indexed by k).

$$z_{v_k} = v_k + \omega_{v_k} \tag{3}$$

We characterize the uncertainty as mutually independent additive, zero-mean, Gaussian white noise.

$$\omega_{v_k} \sim \mathcal{N}(0, \sigma_{v_k}^2) \tag{4}$$

Transforming these sensor frame measurements into a local coordinate frame requires knowledge about sensor and vehicle attitude. Heading is the most important and difficult to accurately observe measurement for this coordinate rotation. Again we use a simple additive Gaussian noise model to represent the heading (ψ) measurement.

$$z_\psi = \psi + \omega_\psi \tag{5}$$

$$\omega_\psi \sim \mathcal{N}(0, \sigma_\psi^2) \tag{6}$$

It is possible to carry forward the complete three dimensional ($k = \{1,2,3\}$) formulation (Eustice, Whitcomb, Singh, & Grund, 2007), but it is non-limiting to simplify this representation to a two dimensional representation. In particular we assume the pitch and roll are transformations that do not affect the uncertainty growth. We also consider the uncertainty along-track to be independent of the uncertainty across track. These considerations capture the dominant dynamics of error growth (velocity and heading uncertainty) and allow us to simplify our two-dimensional model, preserving intuition. The resulting odometry measurement model considers discrete observations of incremental distance ($\boldsymbol{z}_{o_j}$), where j is the temporal index for sequential velocity measurements.

$$\boldsymbol{z}_{o_j} = (\boldsymbol{x}_{h_j} - \boldsymbol{x}_{h_{j-1}}) + \boldsymbol{\omega}_o \tag{7}$$

The additive noise is characterized by a two-dimensional covariance matrix ($\boldsymbol{\Sigma}_\mathrm{o}$) in the along track and across track directions.

$$\boldsymbol{\omega}_o \sim \mathcal{N}(0, \boldsymbol{\Sigma}_\mathrm{o}) \tag{8}$$

$$\boldsymbol{\Sigma}_\mathrm{o} = \begin{bmatrix} t\sigma_v^2 & 0 \\ 0 & d^2\sigma_\psi^2 \end{bmatrix} \tag{9}$$

The diagonal matrix in equation (9) is a consequence of the independent along track and across track uncertainty growth. The along track term, in the upper left, captures growth of position uncertainty as a function of velocity uncertainty, based on random walk uncertainty growth. The across track term, in the lower right, is dominated by heading uncertainty; therefore, the across track uncertainty grows linearly with distance travelled.
Returning to Fig. 6 we can predict how the odometry error will grow for a straight line vehicle trajectory. The figure shows the along track uncertainty the x direction and across track uncertainty in the y direction. The aspect ratio of error ellipses increases with time, illustrating combination of linear growth of the along track uncertainty (growing with distance travelled) and growth proportional to the square root of time of the along track position.

3.4 Data fusion

These two standalone navigation solution, LBL positioning and DVL dead-reckoning, are a complementary pair of information sources. Fusing these sources can exploit both the precision of the DVL solution and the accuracy of the LBL reference. The introduction to this section provided a qualitative discussion of this integration, and there are many excellent references with the details of how to combine these two sensing modalities.
(Whitcomb, Yoerger, & Singh, 1999) (Larsen M. B., 2000).

4. Tradeoffs in designing navigation solutions

How does the vehicle designer decide which navigation solutions to employ and how to configure them? This section describes a framework for making these decisions based on

applications of estimation theory to the problem of estimating position based on noisy measurements. This model enables designers to predict the performance of candidate designs based on their quantitative performance metrics. Using this analysis framework we present the answer to particular questions often asked when designing and deploying a range-based positioning system:

- What is the "best" geometry of the fixed acoustic nodes and mobile nodes in an LBL network? What is the sensitivity of the system precision with respect to changes in this geometry?
- What is the relative importance of geometry vis-á-vis range precision in an LBL network?
- What is the best range-based configuration (geometry and update rate) to integrate with dead-reckoning solutions.

To quantify these tradeoffs we propose metrics for positioning precision based on standard terrestrial positioning problems. We use the Cramér Rao lower bound (CRLB) to frame the question in a way that affords thorough analysis. Based on this framework, we articulate particular design tradeoffs, e.g., how design choices affect precision of the position estimate.

4.1 Analytical framework for predicting performance

Navigation is an estimation problem; a set of unknown parameters, location and attitude, are estimated from a set of observations. The CRLB is a standard tool for determining the uncertainty in the estimate based on uncertainty in the observations and a model relating the observed and estimated quantities.

Consider the estimation of an unknown parameter vector $\boldsymbol{x}$ from a set of observations $\boldsymbol{z}$ with known probability density $p_{\boldsymbol{z}}(\boldsymbol{z};\boldsymbol{x})$. An estimator extracts the information from these observations to derive and estimate of the parameters based on the measurements, $\hat{\boldsymbol{x}}(\boldsymbol{z})$. The uncertainty in this estimate is a direct consequence of how much information is available from the measurements. When it exists, the CRLB gives the lower bound on the variance of *any* valid unbiased estimator (Bar-Shalom, Li, & Kirubarajan, 2001). The *Fisher information*, $\boldsymbol{I}_{\boldsymbol{z}}(\boldsymbol{x})$ is the information about the parameters, $\boldsymbol{x}$ contained in the observations, $\boldsymbol{z}$.

$$\boldsymbol{I}_{\boldsymbol{z}}(\boldsymbol{x}) = E\left[\frac{\delta^2}{\delta x^2}\ln p_{\boldsymbol{z}}(\boldsymbol{z};\boldsymbol{x})\right]\boldsymbol{z}_{o_j} = (\boldsymbol{x}_{h_j} - \boldsymbol{x}_{h_{j-1}}) + \boldsymbol{\omega}_o \tag{10}$$

Where $E[\ ]$ is the expectation operator. The CRLB, $\boldsymbol{\Lambda}(\hat{\boldsymbol{x}}(\boldsymbol{z}))$, is the inverse of the Fisher information, i.e.,

$$\boldsymbol{\Lambda}\big(\hat{\boldsymbol{x}}(\boldsymbol{z})\big) = [\boldsymbol{I}_{\boldsymbol{z}}(\boldsymbol{x})]^{-1} \tag{11}$$

The CRLB is the minimum uncertainty achievable by an unknown optimal estimator. An estimator that approaches this existence of the lower bound is *efficient*, but the bound does not guarantee that an efficient estimator exists or that one can be found. Another consequence of this principle is that an efficient estimator extracts all the available information from the observations. Efficiency amounts to the extracted information being equal to the existing information.

4.1.1 The CRLB for standalone spherical positioning

When LBL positioning is used alone, without other complementary references, the precision of such a solution is based on (1) the precision of the range measurements, (2) the geometry of the fixed transponders and mobile host, (3) the accuracy of estimate of the speed of sound and (4) the uncertainty in the estimated location of the fixed seafloor transponders. We can

consider each of these sources of uncertainty by applying the CRLB framework to the spherical positioning measurement model described in Section 3.2. The range measurements are assembled into an measurement vector of length n.

$$\boldsymbol{Z}_r = \{z_{r_i}\} = \boldsymbol{h}(\boldsymbol{x}_h, \boldsymbol{x}_{\boldsymbol{b}_i}) + \boldsymbol{w}_r \tag{12}$$

where $\boldsymbol{h}(\ )$ is the non-linear function for spherical positioning (equation (1)) and $\boldsymbol{w}_r$ is a zero mean random vector with covariance $\boldsymbol{\Sigma}_{\mathrm{r}}$.

$$\boldsymbol{w}_r \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\mathrm{r}}) \tag{13}$$

The CRLB is calculated by linearizing the measurement model about an operating point, $\boldsymbol{x}_{h_o}$. The result is summarized by the first derivative of the measurement equation evaluated at the operating point, i.e., the Jacobian matrix $\boldsymbol{C}$. For the linearized measurement model with additive Gaussian noise, the CRLB is a matrix combination of the Jacobian, representing the current system geometry, and the measurement covariance quantifying the observation uncertainty.

$$\boldsymbol{\Lambda} = \boldsymbol{C}^T \boldsymbol{\Sigma}_{\mathrm{r}}{}^{-1} \boldsymbol{C} \tag{14}$$

The CRLB is the best-case performance of an unbiased estimator designed to estimate the mobile host position based on uncertain range observations. The CRLB matrix is the minimum value of the covariance matrix for any unbiased estimate of position, i.e.,

$$\boldsymbol{\Lambda} \leq \boldsymbol{\Sigma}_{\mathrm{x_h}} = \boldsymbol{E}\left[[\hat{\boldsymbol{x}}_h - \overline{\boldsymbol{x}}_h][\hat{\boldsymbol{x}}_h - \overline{\boldsymbol{x}}_h]^T\right] \tag{15}$$

where $\overline{\boldsymbol{x}}_h$is the unknown true position of the host and $\hat{\boldsymbol{x}}_h$ is the estimated mobile host position. To summarize, the CRLB is a best-case estimate of the state covariance of the position solution as expressed in based on the geometry of the static acoustic beacons, the location of the host relative to the beacons and the range uncertainty.

4.1.2 The CRLB for combined odometry and positioning

The CRLB framework is also capable of analyzing the tradeoffs inherent in combining observations into an integrated navigation solution. In particular, we are interested in quantifying the tradeoffs involved in combining LBL absolute positioning with DVL dead-reckoning.

To apply the CRLB framework to this case requires a measurement model including both the high update rate odometry measurements of relative distance travelled and infrequent absolute position updates. In one-dimension the absolute position measurement uncertainty is equivalent to the range uncertainty (σ_r). To consider two-dimensional the odometry observation model from equations (7)-(9) we sum the two independent components of uncertainty. This simplification is similar to the notion of scalar horizontal precision discussed in the next section.

$$\sigma_o^2 = \mathrm{t}\,\sigma_v^2 + \mathrm{d}^2\,\sigma_\psi^2 \boldsymbol{\Lambda} \leq \boldsymbol{\Sigma}_{\mathrm{x_h}} = \boldsymbol{E}\left[[\hat{\boldsymbol{x}}_h - \overline{\boldsymbol{x}}_h][\hat{\boldsymbol{x}}_h - \overline{\boldsymbol{x}}_h]^T\right] \tag{16}$$

Now we can create a combined one-dimensional measurement model for a set of n absolute position updates with $n-1$ interspersed odometry measurements.

$$\boldsymbol{Z_c} = \begin{Bmatrix} \mathrm{x_1} \\ \vdots \\ \mathrm{x_n} \\ (\mathrm{x_2} - \mathrm{x_1}) \\ \vdots \\ (\mathrm{x_n} - \mathrm{x_{n-1}}) \end{Bmatrix} + \boldsymbol{\omega}_\mathrm{c} \tag{17}$$

The additive noise vector, $\boldsymbol{\omega}_\mathrm{c}$, is modelled using a zero-mean Gaussian distribution. The individual measurements are considered to be independent, resulting in a covariance matrix that based on the standalone range measurements and odomentry measurements.

$$\omega_c \sim \mathcal{N}(0, \boldsymbol{\Sigma}_\mathrm{c}) \tag{18}$$

$$\boldsymbol{\Sigma}_\mathrm{c} = \begin{bmatrix} \sigma_r^2 \, \boldsymbol{I}_n & 0 \\ 0 & \sigma_o^2 \, \boldsymbol{I}_{n-1} \end{bmatrix} \tag{19}$$

Where $\boldsymbol{I}_n$is an $n \times n$ identity matrix.

4.2 Metrics for positioning performance

Horizontal dilution of precision (HDOP) and *circular error probable* (CEP) provide a quantifiable measure to succinctly convey the positioning precision for design and deployment decisions. The two positioning metrics are based the uncertainty in host (mobile station or vehicle) position estimate. The covariance matrix of the unknown error is the state estimate is

$$\boldsymbol{\Sigma}_{\mathbf{x}_\mathrm{h}} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 & \sigma_{xz}^2 \\ \sigma_{xy}^2 & \sigma_y^2 & \sigma_{yz}^2 \\ \sigma_{xz}^2 & \sigma_{yz}^2 & \sigma_z^2 \end{bmatrix} \tag{20}$$

Because typical positioning geometries afford different performance in the horizontal plane as compared with the vertical dimension, the three-dimensional covariance is often decomposed into the horizontal (2D) and vertical components. The horizontal components (x and y) of the covariance represented an uncertainty ellipses as illustrated in Fig. 6.

4.2.1 Dilution of precision

Dilution of precision metrics are common in GPS applications. The horizontal dilution of precision comes directly from the Cartesian components (x and y) of the position estimate covariance matrix in equation (20).

$$\sigma_{hdop} = \frac{\sqrt{\sigma_x^2 + \sigma_y^2}}{\sigma_r} \tag{21}$$

The HDOP metric is normalized by the range uncertainty (σ_r)to isolate the sensitivity of the metric to the solutions geometry. Fig. 7 illustrates how this Cartesian interpretation overestimates the uncertainty by describing a rectangular boundary of uncertainty ellipse.

4.2.2 Circular error probable

In contrast to the DOP metric, the CEP metric is volumetric and non-normalized. The CEP defines the radius of the smallest circle, centered at the estimate, that has a 50% probability

of containing the true value. A linear approximation of the CEP can be derived from the estimate covariance.

$$CEP \approx 0.59\,(\sigma_L + \sigma_S) \tag{22}$$

where σ_L and σ_S are the major and minor axes of the uncertainty ellipse as shown in Fig. 7. The major and minor axes are the eigenvalues of the two-dimensional covariance matrix. The difference between the true CEP and the approximation of equation (22) is less than 1.5% when the uncertainty ellipse has a low aspect ratio ($0.5\sigma_L \leq \sigma_S \leq \sigma_L 0.5$), otherwise a quadratic approximation should be used (Nelson, 1988).
The CEP metric is volumetric because it uses the principle directions rather than the Cartesian directions, but is not normalized and therefore is a function of both the geometry and the range uncertainty.

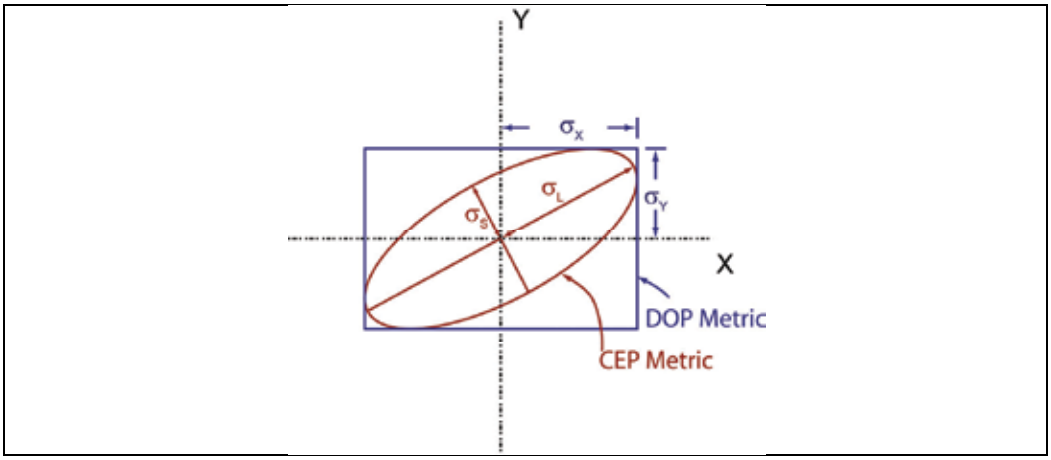


Fig. 7. Illustration of the covariance metrics. Geometrically the 2D covariance can be represented with an ellipse. The diagonal terms of the fully populated 2x2 matrix are σ_x^2 and σ_y^2. The square root of the two eigenvalues are one half major (σ_L) and minor (σ_S) axes of the ellipse. (Figure is adapted from (Kaplan, 1996).)

4.3 Results: Predicting performance metrics using the CRLB

Using the estimation framework of Section 4.1 and the performance metrics from Section 4.2 we can quantify the tradeoffs involved in designing standalone LBL positioning and integrated LBL/DVL navigation.

4.3.1 Standalone LBL configuration

Applying the CRLB to standalone LBL positioning enables the designer to predict the influence of transponder geometry, host location and range uncertainty on the LBL solution. Fig. 8 shows the results of this analysis. To generate these results the CRLB (equation (14)) is evaluated at each point in the two-dimensional space. The figure shows the results for a prototypical configuration, where the transponders are arranged in an equilateral triangle. The specific example shown in Fig. 8 illustrates the general process. This process has proven useful in deciding how to configure a LBL solution or deciding the level of range precision necessary to meet a particular performance specification.

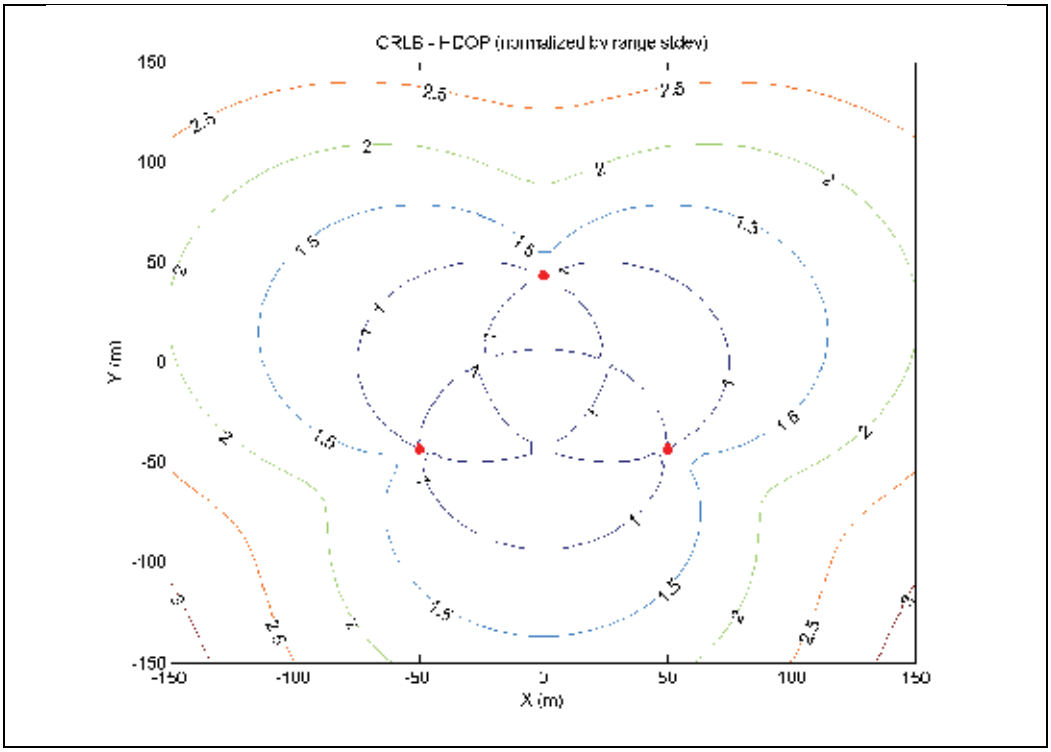


Fig. 8. Positioning performance prediction for standalone LBL positioning. The transponder locations are indicated by the red markers in an equilateral triangle near the origin. The contours show lines of constant HDOP.

4.3.2 Integrated LBL/DVL solution

Building on the standalone analysis of the previous section, next we use the CRLB framework to consider design decisions inherent in combining absolute positioning (LBL) and odometry dead-reckoning (DVL+Heading). The sensing modalities are best used in concert, where the two information sources can complement each other. The CRLB framework, using the measurement model in equation (17), quantifies the benefits of this combination.

The LBL and DVL sensing modalities must be matched to realize the potential of the complementary nature of these two navigation methods. Fig. 9 illustrates the constructive combination of LBL range observations, DVL velocity measurements and heading reference using a simple one-dimensional model. This comparison guides the selection of relative precision of the various sensors and the required update rate to leverage ability of absolute positioning to constrain the drift inherent to dead reckoning.

The two asymptotes in Fig. 9 are illustrative. On the right, in Region 3, we see that as odometry error is large, the overall positioning uncertainty is limited to be approximately equivalent to the absolute positioning uncertainty, indicated by $\sigma_o \,/\, \sigma_r \cong 1.0$ when the dead-reckoning uncertainty is greater than twice the absolute uncertainty ($\sigma_o > 2.0\, \sigma_r$). This could be caused by either high uncertainty in the velocity or heading measurements or large update times between absolute position updates. Conversely, the left side of the figure, Region 1, shows how precise odometry between absolute position updates links the sequential updates together. As the odometry becomes more precise the overall position

uncertainty approaches the bound of $\sigma_x = \sigma_r/\sqrt{N}$, where N is the number of discrete position updates (in this case $N = 100$). This limiting case represents perfect odometry, where the distance between absolute reference updates is known.

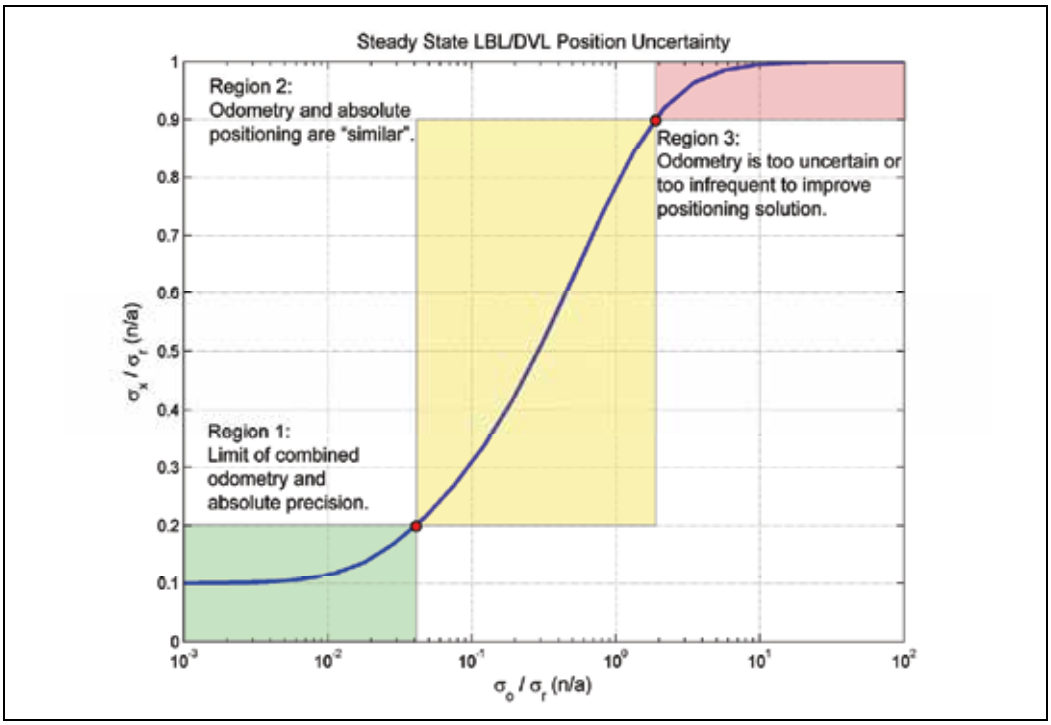


Fig. 9. Based on the one-dimensional model, this figure quantifies the tradeoffs in designing a complementary positioning solution using absolute positioning (LBL) and dead-reckoning odometry (DVL+Heading). The vertical axis shows position uncertainty (σ_x) normalized by the absolute reference uncertainty (σ_r). The horizontal axis shows the ratio of total odometry uncertainty (σ_o) to absolute reference uncertainty. Designing a solution in Region 1, with $\frac{\sigma_o}{\sigma_r} < 0.04$, successfully leverages the complementary nature of the two modes of navigation.

As an illustration we present an example using representative numbers for instruments typical on modern underwater platforms. Based on Fig. 9 we would like to design the positioning solution to operate in Region 1, where the total odometry uncertainty is less then 0.04 times the absolute positioning uncertainty, i.e.,

$$\sigma_o = \sigma_v \sqrt{t} + d\, \sigma_\psi < 0.1\, \sigma_r \tag{23}$$

Typical vehicle instrumentation might consist of an 1,200 kHz RDI DVL[3] ($\sigma_v = 3$ mm/s), an Octans true north heading reference[4] ($\sigma_\psi = 0.1$ degrees) and Benthos LBL transponders[5] ($\sigma_r \cong 3.0$ m). Furthermore we can assume a typical velocity of 1.0 m/s for the purposes of demonstration, resulting in $d = 1.0\, t$. Therefore,

[3] 1,200 kHz Workhourse Navigator Doppler velocity log by Teledyne RD Instruments.
[4] 6000 Series Transponders by Teledyne Benthos.
[5] Octans Fiber Optic Gyroscope (FOG) by Ixsea.

$$0.003\sqrt{t} + 0.0035\,t < 0.04\ (3.0) \tag{24}$$

Resulting in a required update rate of $t < 56$ seconds. Such an infrequent update rate is a consequence of the precision of the dead-reckoning solution.

5. Next steps: Opportunities to improve AUV navigation

Navigation continues to limit the application of AUV technology. As AUVs continue to be adopted by new users for new applications the fundamental navigation challenges must be addressed to further this expansion. Starting from the current state of the art we attempt to identify a few fruitful areas for continued research and development; areas that promise to have a strong impact on the design of new AUV systems.

5.1 Crossing chasms

There are gaps in current AUV navigation capabilities. Two of these gaps are explored below along with possible directions aimed at closing these gaps.

5.1.1 Decreasing transponder dependence

Many efforts in navigation research and development seek to reduce (or eliminate) the role of seafloor moored transponders in a navigation solution. As discussed above, LBL transponders provide an absolute reference, but this comes at a high cost. Transponders are deployed and surveyed from the surface in preparation for AUV missions and then recovered after completion of the mission. This evolution erodes operational efficiency, requiring hours or even days to complete depending on the environment and the mission. Such a seafloor-based external reference also limits the range of an AUV; typical transponder networks can only cover a few square kilometres.

Work has been done to eliminate the survey step in deploying transponders. One solution is to place the transponders at the surface, on floating buoys, where GPS can provide constant position updates. This has been used for tracking (the position is recorded at the surface, but not available subsea in real-time) AUVs for survey operations (Desset, Damus, Morash, & Bechaz, 2003). Another approach is to concurrently localize the fixed transponders while navigating using the range information. Using concurrent localization and mapping (CML), also known as simultaneous localization and mapping (SLAM), researchers have created a consistent map of the environment using only range information when the transponder locations are not known before the mission (Olson, Leonard, & Teller, 2006). Yet another approach is to have the AUV actually deploy the fixed transponders. This solution addresses a military need to limit the detection for AUV operations.

Instead of reducing the time spent on survey, another possible method is to decrease the number of transponders necessary to provide an absolute reference. Initial research efforts were focused on proving the theory of single beacon navigation (Larsen M. B., 2000). More recently this effort has moved from theoretical research to practical implementation, including algorithm development and integration into operational platforms such as the REMUS AUV (Hartsfield, 2005).

The incorporation of reliable acoustic communication has provided additional opportunity solutions to decrease the dependence on acoustic transponders. With the ability to transmit ephemeris data from a surface ship to the submerged platform, it becomes possible to eliminate the transponders all together and use the moving surface ship (with GPS navigation)

as fixed reference (Eustice, Whitcomb, Singh, & Grund, 2007). The Hugin AUV, a successful commercial survey tool, has used a similar technique to provide position updates and change the AUV mission from the surface (Vestgard, Storkersen, & Sortland, 1999)

5.1.2 Sensors: payload versus navigation

Current systems differentiate between navigation sensors and payload sensors. Navigation sensors are specifically for collecting measurements to position an AUV. These observations are processed in real-time using a variety of perception algorithms. In contrast, payload instruments collect data for future processing. Powerful instruments such a multibeam sonar, high quality still cameras, etc. are used to collect high resolution data about the environment, but this information is not used in real-time.

Many projects are seeking to alleviate this divide between payload and navigation sensors. Vision based algorithms promise to leverage the optical images to constrain the unbounded error growth for underwater applications (Huster & Rock, 2003) (Eustice, Pizarro, & Singh, 2004). Similarly, combining course navigation with bathymetry can serve to improve both the positioning and final data product (Roman & Singh, 2006). Many researchers have developed estimation techniques that make use of the bathymetry. These terrain based methods make use of either a fathometer or bathymetric sonar to position the vehicle relative to a known (or partially unknown) map of the seafloor (Tuohy, Leonard, Bellingham, Patrikalakis, & Chryssostomidis, 1996) (Williams, Dissanayake, & Durrant-Whyte, 1999). Each of these techniques offers a path toward crossing the artificial divide between payload sensors and navigation aids.

5.2 Operations in challenging environments

The application of AUV technology for exploration and investigation is moving into new environments. This valuable technology has improved our ability to accomplish nearbottom surveys in the open ocean. Now the needs of new users are necessitating adaptation of AUV technologies to a variety of interesting and challenging underwater environments. Under-ice missions promise to open the important polar regions to the observational power of autonomous platforms (Kunz, et al., 2008) (McEwen, Thomas, Weber, & Psota, 2005). Obviously navigation under-ice is very important to the safety of such mission; the AUV must be able to return a safe region for recovery. Possibly not so obvious are the challenges presented by the acoustics of under-ice environments. The upward refracting acoustic environment can create shadow zones, restricting the means of communication and positioning. Furthermore the ice cover can shift at significant speeds relative to the seafloor, creating dynamic environment for navigation (von der Heydt, Duckworth, & Baggeroer, 1985) (Deffenbaugh, Schmidt, & Bellingham, 1993).

Another environment that presents new challenges for AUV operations is coastal zones. The littoral zone has been recognized by military users as a key new frontier for operations. Similarly, environmental assessment of shallow marine environments (e.g., coral reefs) is pushing AUV missions towards the coast. From a navigation perspective, these shallow water environments can be more dynamic than the deep ocean with increased multipath and high background noise from breaking waves and other disturbances.

A last example of new environments for AUV operations is the exploration of freshwater caves, cenotes, using novel AUVs. Research expeditions have used three dimensional SLAM-based to map these underwater caves. Interestingly, these expeditions are supported by resources for space exploration because of the analogy between cenote exploration and the environment operators anticipate for autonomous exploration of other planets (Kumagni, 2007).

6. Continued improvement in AUV navigation

One way to set expectations for the future is to look at the past. In the past two decades of AUV platform development autonomous navigation has provided fundamental supporting technology through new instruments, new algorithms and new methods of operation. As AUV platforms continue to proliferate, becoming commercially available to a wider user base, we can expect the opportunities for improved navigation methods to similarly expand. Operators and vehicle designers will need new solutions that increase efficiency, decrease cost and allow for the application of AUV technology to exciting new environments.

7. Acknowledgements

The Deep Submergence Lab at Woods Hole Oceanographic provided the data and images used in developing the illustrations in Fig. 2 and Fig. 3. Also, Joel Gendron illustrated Fig. 4. The publication of this chapter was supported by the Franklin W. Olin College of Engineering in Needham, Massachusetts.

8. References

Bar-Shalom, Y., Li, X. -R., & Kirubarajan, T. (2001). *Estimation with applications to tracking and navigation.* John Wiley and Sons, Inc.

Bingham, B., Mindell, D., Wilcox, T., & Bowen, A. (2006). Integrating precision relative positioning into JASON/MEDEA ROV operations. *Marine Technology Society (MTS) Journal, 40* (1), pp. 87-96.

Catipovic, J. A. (1990). Performance limitations in underwater acoustic telemetry. *IEEE Journal of Oceanic Engineering, 15* (3), pp. 205-216.

Deffenbaugh, M., Schmidt, H., & Bellingham, J. G. (1993). Acoustic navigation for Arctic under-ice AUV missions. *OCEANS '93. Proceedings.*

Desset, S., Damus, R., Morash, J., & Bechaz, C. (2003). Use of GIBs in AUVs for underwater archaeology. *Sea Technology.*

Dunlap, G. D. (1975). *Dutton's Navigation and Piloting.* Unites States Naval Institute Press.

Eustice, R. M., Singh, H., & Leonard, J. J. (2006). Exactly sparce delayed-state filters for view-based SLAM. *IEEE Transactions on Robotics, 22* (6), pp. 1100-1114.

Eustice, R. M., Whitcomb, L. L., Singh, H., & Grund, M. (2007). Experimental results in synchronous-clock one-way-travel-time acoustic navigation for autonomous underwater vehicles. *Proc. of the IEEE International Conference on Robotics and Automation*, 4257-4264.

Eustice, R., Pizarro, O., & Singh, H. (2004). Visually augmented navigation in an unstructured environment using a delayed state history. *Proc. of the IEEE International Conference on Robotics and Automation, 1*, 25-32.

Frew, E. W., & Rock, S. M. (2003). Trajectory generation for monocular-vision based tracking of a constant-velocity target. *Proc. of the IEEE International Conference on Robotics and Automation.* Taipei, Taiwan.

Hartsfield, J. C. (2005). *Single Transponder Range Only Navigation Geometry (STRONG) Applied to REMUS Autonomous Under Water Vehicles.* Masters Thesis, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution.

Hunt, M. M., Marquet, W. M., Moller, D. A., Peal, K. R., Smith, W. K., & Spindel, R. C. (1974). *An acoustic navigation system.* Technical Report WHOI-74-6, Woods Hole Oceanographic Institution.

Huster, A., & Rock, S. M. (2003). Relative position sensing by fusing monocular vision and inertial rate sensors. *Proc. of the 11th International Conference on Advanced Robotics*, *3*, pp. 1562-1567. Coimbra, Portugal.

Huster, A., Fleischer, S. D., & Rock, S. M. (1998). Demonstration of a vision-based dead-reckoning system for navigation of an underwater vehicle. *Proc. of the MTS/IEEE Oceans Conference*, (pp. 326-330).

Kaplan, D. (1996). *Understanding GPS.* Artech House Publisher.

Kinsey, J. C., Smallwood, D. A., & Whitcomb, L. L. (2003). A new hydrodynamics test facility for UUV dynamics and control research. *Proc. of the MTS/IEEE Oceans Conference.* San Diego, CA.

Kumagni, J. (2007). Swimming to Europa. *Spectrum, IEEE , 44* (9), pp. 33-40.

Kunz, C., Murphy, C., Camilli, R., Sing, H., Bailey, J., Eustice, R., et al. (2008). Deep sea underwater robotic exploration in the ice-covered Arctic ocean with AUVs. Proc. of *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (pp. 3654-3660).

Larsen, M. B. (2000). High performance Doppler-inertial navigation - experimental results. *Proc. of the MTS/IEEE Oceans Conference.*

Larsen, M. B. (2000). Synthetic long baseline navigation of underwater vehicles. *Proc. of the MTS/IEEE Oceans Conference*, *3*, pp. 2043-2050.

McEwen, R., Thomas, H., Weber, D., & Psota, F. (2005). Performance of an AUV navigation system at Arctic latitudes. *IEEE Journal of Oceanic Engineering, 30* (2), pp. 443-454.

Milne, P. H. (1983). *Underwater Acoustic Positioning Systems.* Houston: Gulf Publishing Company.

Nelson, W. (1988). *Use of circular error probability in target detection.* United States Air Force Hanscom Air Force Base: MITRE Corporation.

Olson, E., Leonard, J. J., & Teller, S. (2006). Robust range-only beacon localization. *IEEE Journal of Oceanic Engineering, 31* (4), pp. 949-958.

Roman, C., & Singh, H. (2006). Consistency based error evaluation for deep sea bathymetric mapping with robotic vehicles. *Proc. of IEEE International Conference on Robotics and Automation*, pp. 3569-3574.

Smith, S. M., & Kronen, D. (1997). Experimental results of an inexpensive short baseline acoustic positioning system for AUV navigation. *Proc. of the MTS/IEEE Oceans Conference, 1*, pp. 714-720.

Tuohy, S. T., Leonard, J. J., Bellingham, J. G., Patrikalakis, N. M., & Chryssostomidis, C. (1996). Map based navigation for autonomous underwater vehicles. *International Journal of Offshore and Polar Engineering*.

Vestgard, K., Storkersen, N., & Sortland, J. (1999). Seabed surveying with Hugin AUV. *Proc. of the 11th International Symposium on Unmanned Untethered Submersible Technology.* Durham, NH.

von der Heydt, K., Duckworth, G., & Baggeroer, A. (1985). Acoustic array sensor tracking system. *Proc. of the MTS/IEEE Oceans Conference, 17*, pp. 464-471.

Whitcomb, L. L., Yoerger, D. R., & Singh, H. (1999). Combined Doppler/LBL based navigation of underwater vehicles. *Proc. Int. Symp. on Unmanned Untethered Submersible Technology.*

Williams, S. B., Dissanayake, G., & Durrant-Whyte, H. (1999). Towards terrain-aided navigation for underwater robotics. *Advanced Robotics, 15* (5), pp. 533-549.

4

Guidance Laws for Autonomous Underwater Vehicles

Morten Breivik[1] and Thor I. Fossen[1,2]
[1]*Centre for Ships and Ocean Structures*
[2]*Department of Engineering Cybernetics*
Norwegian University of Science and Technology
Norway

1. Introduction

About 70% of the surface of the Earth is covered by oceans, and the ocean space represents a vast chamber of natural resources. In order to explore and utilize these resources, humankind depends on developing and employing underwater vehicles, not least unmanned underwater vehicles (UUVs). Today, UUVs encompass remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs).

The first ROVs were built in the 1950s, put into commercial use in the 1980s, and are mostly used today by the offshore oil and gas industry to carry out inspection and intervention operations at subsea installations (Antonelli et al. 2008). These vehicles are teleoperated by connection to a surface vessel through an umbilical cable that provides them with power and telemetry. In particular, the dependence on a tether represents a considerable challenge for ROV deepwater operations (Whitcomb 2000).

On the other hand, AUVs are free-swimming vehicles that rely on their own energy supply. The first AUVs were built in the 1970s, put into commercial use in the 1990s, and today are mostly used for scientific, commercial, and military mapping and survey tasks (Blidberg 2001). Developed in cooperation between Kongsberg Maritime and the Norwegian Defence Research Establishment, the HUGIN series represents the most commercially successful AUV series on the world market today (Hagen et al. 2003). HUGIN vehicles have been employed for commercial applications since 1997 and for military applications since 2001. The workhorse HUGIN 3000 has an impressive 60 hours endurance at 4 knots speed with payload sensors running. Currently, the main challenges for AUVs encompass endurance, navigation, communication, and autonomy issues.

Traditionally, ROVs and AUVs have been assigned different tasks due to different strengths and weaknesses, see Fig. 1. In the future, hybrid ROV/AUV designs are expected to bridge the gap between these two main UUV types, utilizing the best of both worlds (Wernli 2000).

Regarding motion control research for UUVs, Craven et al. (1998) give an overview of modern control approaches with an emphasis on artificial intelligence techniques; Roberts & Sutton (2006) treat guidance, navigation, and control issues for unmanned marine vehicles with an emphasis on underwater vehicles; while Antonelli et al. (2008) present a state-of-the-art survey of control-related aspects for underwater robotic systems.

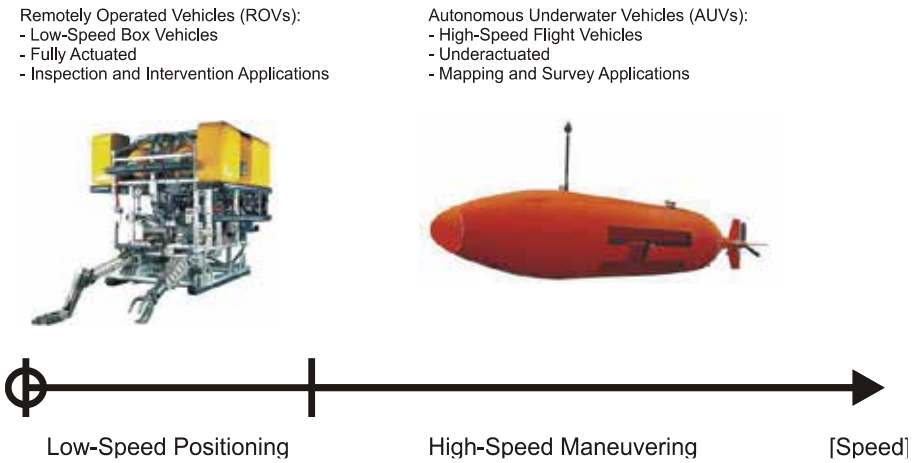


Fig. 1. The two traditional types of UUVs: ROVs and AUVs. These vehicles have different designs and perform different operations in different parts of the speed regime

An essential quality for free-swimming underwater vehicles like AUVs is their ability to maneuver accurately in the ocean space. Motion control is a fundamental enabling technology for such a quality, and every motion control system requires a guidance component. This guidance requirement serves as the main motivation for this work, whose aim is to provide a convenient overview of guidance laws applicable to motion control of AUVs. An extension of (Breivik & Fossen 2008), the exposition is deliberately kept at a basic level to make it accessible for a wide audience. Details and proofs can be found in the references.

1.1 Guidance

According to Shneydor (1998), guidance is defined as: *The process for guiding the path of an object towards a given point, which in general may be moving*. Also, the father of inertial navigation, Charles Stark Draper, states in (Draper 1971) that: *Guidance depends upon fundamental principles and involves devices that are similar for vehicles moving on land, on water, under water, in air, beyond the atmosphere within the gravitational field of earth and in space outside this field*, see Fig. 2. Thus, guidance represents a basic methodology concerned with the transient motion behavior associated with the achievement of motion control objectives.

The most rich and mature literature on guidance is probably found within the guided missile community. In one of the earliest texts on the subject (Locke 1955), a guided missile is defined as: *A space-traversing unmanned vehicle which carries within itself the means for controlling its flight path.* Today, most people would probably think about unmanned aerial vehicles (UAVs) when hearing this definition. However, guided missiles have been operational since World War II (Spearman 1978), and thus organized research on guidance theory has been conducted almost as long as organized research on control theory. The continuous progress in missile hardware and software technology has made increasingly advanced guidance concepts feasible for implementation. Today, missile guidance theory encompass a broad spectrum of guidance laws, namely: classical guidance laws; optimal guidance laws; guidance laws based on fuzzy logic and neural network theory; differential-geometric guidance laws; and guidance laws based on differential game theory.

Fig. 2. Fundamental guidance principles apply from subsea to space

As already mentioned, a classical text on missile guidance concepts is (Locke 1955), while more recent work include (Lin 1991), (Shneydor 1998), (Zarchan 2002), (Siouris 2004), and (Yanushevsky 2008). Relevant survey papers include (Pastrick et al. 1981), (Cloutier et al. 1989), (Lin & Su 2000), and (White & Tsourdos 2001). Also, very interesting personal accounts of the guided missile development during and after World War II can be found in (Haeussermann 1981), (Battin 1982), and (Fossier 1984), while MacKenzie (1990) and Westrum (1999) put the development of guided missile technology into a larger perspective.

The fundamental nature and diverse applicability of guidance principles can be further illustrated through a couple of examples. In nature, some predators are able to conceal their pursuit of prey by resorting to so-called motion camouflage techniques (Mizutani et al. 2003). They adjust their movement according to their prey so that the prey perceive them as stationary objects in the environment. These predators take advantage of the fact that some creatures detect the lateral motion component relative to the predator-prey line of sight far better than the longitudinal component. Hence, approaching predators can appear stationary to such prey by minimizing the relative lateral motion, only changing in size when closing in for the kill. Interestingly, this behavior can be directly related to the classical guidance laws from the missile literature (Justh & Krishnaprasad 2006). Also, such guidance laws have been successfully applied since the early 1990s to avoid computationally-demanding optimization methods associated with motion planning for robot manipulators operating in dynamic environments (Piccardo & Honderd 1991).

2. Motion control fundamentals

This section reviews some basic motion control concepts, including operating spaces, vehicle actuation properties, motion control scenarios, as well as the motion control hierarchy. It concludes with some preliminaries.

2.1 Operating spaces

It is useful to distinguish between different types of operating spaces when considering vehicle motion control, especially since such characterizations enable purposeful definitions of various motion control scenarios. The two most fundamental operating spaces to consider are the *work space* and the *configuration space*.

The work space, also known as the operational space (Sciavicco & Siciliano 2002), represents the physical space (environment) in which a vehicle moves. For a car, the work space is 2-dimensional (planar position), while it is 3-dimensional (spatial position) for an aircraft. Thus, the work space is a position space which is common for all vehicles of the same type.

The configuration space, also known as the joint space (Sciavicco & Siciliano 2002), is constituted by the set of variables sufficient to specify all points of a (rigid-body) vehicle in the work space (LaValle 2006). Thus, the configuration of a car is given by its planar position and orientation, while the configuration of an aircraft is given by its spatial position and attitude.

2.2 Vehicle actuation properties

Every variable associated with the configuration of a vehicle is called a degree of freedom (DOF). Hence, a car has 3 degrees of freedom, while an aircraft has 6 degrees of freedom.

The type, amount, and distribution of vehicle thrust devices and control surfaces, hereafter commonly referred to as actuators, determine the actuation property of a vehicle. We mainly distinguish between two qualitatively different actuation properties, namely *full actuation* and *underactuation*. A fully actuated vehicle is able to independently control all its DOFs simultaneously, while an underactuated vehicle is not. Thus, an underactuated vehicle is generally unable to achieve arbitrary tasks in its configuration space. However, it will be able to achieve tasks in the work space as long as it can freely project its main thrust in this space, e.g., through a combination of thrust and attitude control. In fact, this principle is the mode by which most vehicles that move through a fluid operate, from missiles to ships. Even if these vehicles had the ability to roam the work space with an arbitrary attitude, this option would represent the least energy-efficient alternative.

2.3 Motion control scenarios

In the traditional control literature, motion control scenarios are typically divided into the following categories: *point stabilization*, *trajectory tracking*, and *path following*. More recently, the concept of *maneuvering* has been added to the fold as a means to bridge the gap between trajectory tracking and path following (Skjetne et al. 2004). These scenarios are often defined by motion control objectives that are given as configuration-space tasks, which are best suited for fully actuated vehicles. Also, the scenarios typically involve desired motion that has been defined apriori in some sense. Little seems to be reported about tracking of target points for which only instantaneous motion information is available.

However, in this work, both apriori and non-apriori scenarios are considered, and all the motion control objectives are given as work-space tasks. Thus, the scenarios cover more

broadly, and are also suited for underactuated vehicles. The considered scenarios are defined in the following.

The control objective of a *target-tracking scenario* is to track the motion of a target that is either stationary (analogous to point stabilization) or that moves such that only its instantaneous motion is known, i.e., such that no information about the future target motion is available. Thus, in this case it is impossible to separate the spatio-temporal constraint associated with the target into two separate constraints.

In contrast, the control objective of a *path-following scenario* is to follow a predefined path, which only involves a spatial constraint. No restrictions are placed on the temporal propagation along the path.

However, the control objective of a *path-tracking scenario* is to track a target that moves along a predefined path (analogous to trajectory tracking). Consequently, it is possible to separate the target-related spatio-temporal constraint into two separate constraints. Still, this scenario can be viewed as a target-tracking scenario and handled with target-tracking methods, thus disregarding any apriori path information that is available.

Finally, the control objective of a *path-maneuvering scenario* is to employ knowledge about vehicle maneuverability to feasibly negotiate (or somehow optimize the negotiation of) a predefined path. As such, path maneuvering represents a subset of path following, but is less constrained than path tracking since spatial constraints always take precedence over temporal constraints. Path-maneuvering methods can also be used to handle path-tracking scenarios.

2.4 Motion control hierarchy

A vehicle motion control system can be conceptualized to involve at least three levels of control in a hierarchical structure, see Fig. 3. This figure illustrates the typical components of a marine motion control system, encompassing strategic, tactical, and execution levels of control (Valavanis et al. 1997). All the involved building blocks represent autonomy-enabling technology, but more instrumentation and additional control levels are required to attain fully autonomous operation. An example involves collision avoidance functionality, which demands additional sense and avoid components.

This work is mainly concerned with the highest (strategic) control level of Fig. 3. Termed the kinematic control level, it is responsible for prescribing vehicle velocity commands needed to achieve motion control objectives in the work space. Thus, in this work, kinematic control is equivalent to work-space control, and *kinematic controllers* are referred to as *guidance laws*. This level purely considers the geometrical aspects of motion, without reference to the forces and moments that generate such motion.

Next, the intermediate (tactical) level encompass *kinetic controllers*, which do consider how forces and moments generate vehicle motion. These controllers are typically designed by model-based methods, and must handle both parametric uncertainties and environmental disturbances. For underactuated vehicles, they must actively employ the vehicle attitude as a means to adhere to the velocities ordered by the guidance module. The intermediate control level also contains a *control allocation* block which distributes the kinetic control commands among the various vehicle actuators.

Finally, the lowest (execution) level is constituted by the individual *actuator controllers*, which ensure that the actuators behave as requested by the intermediate control module, and ultimately that the vehicle moves as prescribed by the guidance laws.

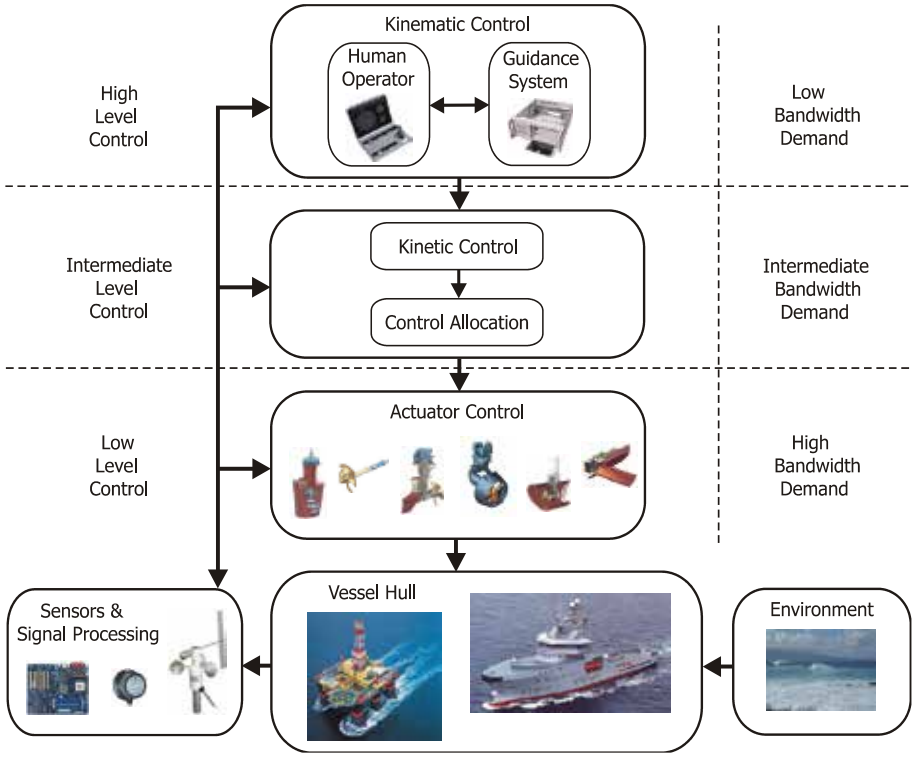


Fig. 3. The motion control hierarchy of a marine surface vessel

2.5 Preliminaries

In the missile literature, guidance laws are typically synonymous with steering laws, assuming that the speed is constant. In this work, guidance laws are either directly prescribed as velocity assignments or partitioned into separate speed and steering laws.

The guidance laws are first introduced in a 2-dimensional framework, where a kinematic vehicle is represented by its planar position $\mathbf{p}(t) \triangleq [x(t), y(t)]^{\mathrm{T}} \in \mathbb{R}^2$ and velocity $\mathbf{v}(t) \triangleq \mathrm{d}\mathbf{p}(t)/\mathrm{d}t \triangleq \dot{\mathbf{p}}(t) \in \mathbb{R}^2$, stated relative to some stationary reference frame. Since most of the AUVs of today are of the survey type, they do not need to perform spatially coupled maneuvers, but typically execute temporally separated planar maneuvers either in the horizontal plane or the vertical plane. Thus, Section 3 and 4 are relevant for such applications. Similar considerations justify the work reported in (Healey & Lienard 1993), (Caccia et al. 2000), and (Lapierre et al. 2003).

In Section 5, the planar methods are extended to a 3-dimensional framework, where a kinematic vehicle is represented by its spatial position $\mathbf{p}(t) \in \mathbb{R}^3$ and velocity $\mathbf{v}(t) \in \mathbb{R}^3$. Results on spatially coupled motion control of AUVs can be found in (Encarnação & Pascoal 2000), (Do & Pan 2003), (Aguiar & Hespanha 2004), (Breivik & Fossen 2005a), (Børhaug & Pettersen 2006), and (Refsnes et al. 2008).

Finally, note that all the illustrations of guidance principles employ the marine convention of a right-handed coordinate system whose z-axis points down.

3. Guidance laws for target tracking

In this section, guidance laws for target tracking are presented. The material is adapted from (Breivik & Fossen 2007).

Denoting the position of the target by $\mathbf{p}_t(t) \triangleq [x_t(t), y_t(t)]^T \in \mathbb{R}^2$, the control objective of a target-tracking scenario can be stated as

$$\lim_{t\to\infty} (\mathbf{p}(t) - \mathbf{p}_t(t)) = \mathbf{0}, \tag{1}$$

where $\mathbf{p}_t(t)$ is either stationary or moving by a (non-zero and bounded) velocity $\mathbf{v}_t(t) \triangleq \dot{\mathbf{p}}_t(t) \in \mathbb{R}^2$.

Concerning tracking of moving targets, the missile guidance community probably has the most comprehensive experience. They commonly refer to the object that is supposed to destroy another object as either a missile, an interceptor, or a pursuer. Conversely, the threatened object is typically called a target or an evader. Here, the designations interceptor and target will be used.

An interceptor typically undergoes 3 phases during its operation; a launch phase, a midcourse phase, and a terminal phase. The greatest accuracy demand is associated with the terminal phase, where the interceptor guidance system must compensate for the accumulated errors from the previous phases to achieve a smallest possible final miss distance to the target. Thus, 3 terminal guidance strategies will be presented in the following, namely line of sight, pure pursuit, and constant bearing. The associated geometric principles are illustrated in Fig. 4.

Note that while the main objective of a guided missile is to hit (and destroy) a physical target in finite time, we recognize the analogy of hitting (converging to) a virtual target asymptotically, i.e., the concept of asymptotic interception, as stated in (1).

3.1 Line of sight guidance

Line of sight (LOS) guidance is classified as a so-called three-point guidance scheme since it involves a (typically stationary) reference point in addition to the interceptor and the target. The LOS denotation stems from the fact that the interceptor is supposed to achieve an intercept by constraining its motion along the line of sight between the reference point and the target. LOS guidance has typically been employed for surface-to-air missiles, often mechanized by a ground station which illuminates the target with a beam that the guided missile is supposed to ride, also known as beam-rider guidance. The LOS guidance principle is illustrated in Fig. 4, where the associated velocity command is represented by a vector pointing to the left of the target.

3.2 Pure pursuit guidance

Pure pursuit (PP) guidance belongs to the so-called two-point guidance schemes, where only the interceptor and the target are considered in the engagement geometry. Simply put, the interceptor is supposed to align its velocity along the line of sight between the interceptor and the target. This strategy is equivalent to a predator chasing a prey in the animal world, and very often results in a tail chase. PP guidance has typically been employed for air-to-surface missiles. The PP guidance principle is represented in Fig. 4 by a vector pointing directly at the target.

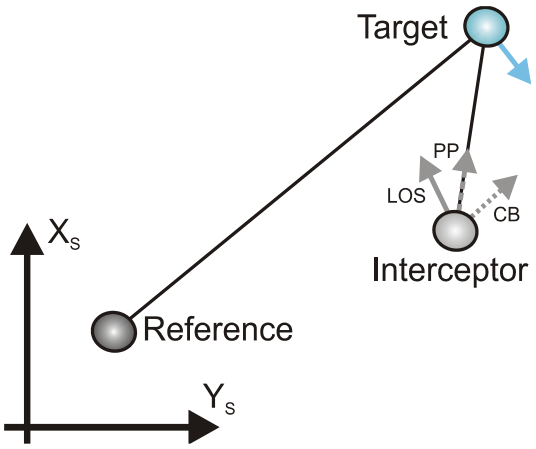


Fig. 4. The interceptor velocity commands that are associated with the classical guidance principles line of sight (LOS), pure pursuit (PP), and constant bearing (CB)

Deviated pursuit guidance is a variant of PP guidance where the velocity of the interceptor is supposed to lead the interceptor-target line of sight by a constant angle in the direction of the target movement. An equivalent term is fixed-lead navigation.

3.3 Constant bearing guidance

Constant bearing (CB) guidance is also a two-point guidance scheme, with the same engagement geometry as PP guidance. However, in a CB engagement, the interceptor is supposed to align the *relative* interceptor-target velocity along the line of sight between the interceptor and the target. This goal is equivalent to reducing the LOS rotation rate to zero such that the interceptor perceives the target at a constant bearing, closing in on a direct collision course. CB guidance is often referred to as parallel navigation, and has typically been employed for air-to-air missiles. Also, the CB rule has been used for centuries by mariners to avoid collisions at sea; steering away from a situation where another vessel approaches at a constant bearing. Thus, guidance principles can just as well be applied to avoid collisions as to achieve them. The CB guidance principle is indicated in Fig. 4 by a vector pointing to the right of the target.

The most common method of implementing CB guidance is to make the rotation rate of the interceptor velocity directly proportional to the rotation rate of the interceptor-target LOS, which is widely known as proportional navigation (PN).

CB guidance can also be implemented through the direct velocity assignment

$$\mathbf{v}(t) = \mathbf{v}_t(t) - \kappa(t)\frac{\tilde{\mathbf{p}}(t)}{|\tilde{\mathbf{p}}(t)|}, \tag{2}$$

where

$$\tilde{\mathbf{p}}(t) \triangleq \mathbf{p}(t) - \mathbf{p}_t(t) \tag{3}$$

is the line of sight vector between the interceptor and the target, $|\tilde{\mathbf{p}}(t)| \triangleq \sqrt{\tilde{\mathbf{p}}(t)^T \tilde{\mathbf{p}}(t)} \geq 0$ is the Euclidean length of this vector, and where $\kappa(t) \geq 0$ can be chosen as

$$\kappa(t) = U_{\mathrm{a,max}} \frac{\left|\tilde{\mathbf{p}}(t)\right|}{\sqrt{\tilde{\mathbf{p}}(t)^{\mathrm{T}}\tilde{\mathbf{p}}(t) + \Delta_{\tilde{\mathrm{p}}}^2}}, \tag{4}$$

where $U_{\mathrm{a,max}} > 0$ specifies the maximum approach speed toward the target, and $\Delta_{\tilde{\mathrm{p}}} > 0$ affects the transient interceptor-target rendezvous behavior.

Note that CB guidance becomes equal to PP guidance for a stationary target, i.e., the basic difference between the two guidance schemes is whether the target velocity is used as a kinematic feedforward or not.

Returning to the example on motion camouflage, it seems that two main strategies are in use; camouflage against an object close by and camouflage against an object at infinity. The first strategy clearly corresponds to LOS guidance, while the second strategy equals CB guidance since it entails a non-rotating predator-prey line of sight.

4. Guidance laws for path scenarios

In this section, guidance laws for different path scenarios are considered, including path following, path tracking, and path maneuvering. Specifically, the guidance laws are composed of speed and steering laws, which can be combined in various ways to achieve different motion control objectives. The speed is denoted $U(t) \triangleq |\mathbf{v}(t)| = \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2} \geq 0$, while the steering is denoted $\chi(t) \triangleq \mathrm{atan2}(\dot{y}(t), \dot{x}(t)) \in \mathbb{S} \triangleq [-\pi, \pi]$, where $\mathrm{atan2}(y, x)$ is the four-quadrant version of $\arctan(y/x) \in \langle -\pi/2, \pi/2 \rangle$.

Path following is ensured by proper assignments to $\chi(t)$ as long as $U(t) > 0$ since the scenario only involves a spatial constraint, while the spatio-temporal path-tracking and path-maneuvering scenarios both require explicit speed laws in addition to the steering laws. The following material is adapted from (Breivik & Fossen 2004a), (Breivik & Fossen 2005b), and (Breivik et al. 2008).

4.1 Steering laws for straight lines

Consider a straight-line path implicitly defined by two waypoints through which it passes. Denote these waypoints as $\mathbf{p}_{\mathrm{k}} \triangleq [x_{\mathrm{k}}, y_{\mathrm{k}}]^{\mathrm{T}} \in \mathbb{R}^2$ and $\mathbf{p}_{\mathrm{k+1}} \triangleq [x_{\mathrm{k+1}}, y_{\mathrm{k+1}}]^{\mathrm{T}} \in \mathbb{R}^2$, respectively. Also, consider a path-fixed reference frame with origin in $\mathbf{p}_{\mathrm{k}}$, whose x-axis has been rotated a positive angle $\alpha_{\mathrm{k}} \triangleq \mathrm{atan2}(y_{\mathrm{k+1}} - y_{\mathrm{k}}, x_{\mathrm{k+1}} - x_{\mathrm{k}}) \in \mathbb{S}$ relative to the x-axis of the stationary reference frame. Hence, the coordinates of the kinematic vehicle in the path-fixed reference frame can be computed by

$$\boldsymbol{\varepsilon}(t) = \mathbf{R}(\alpha_{\mathrm{k}})^{\mathrm{T}}(\mathbf{p}(t) - \mathbf{p}_{\mathrm{k}}), \tag{5}$$

where

$$\mathbf{R}(\alpha_{\mathrm{k}}) \triangleq \begin{bmatrix} \cos\alpha_{\mathrm{k}} & -\sin\alpha_{\mathrm{k}} \\ \sin\alpha_{\mathrm{k}} & \cos\alpha_{\mathrm{k}} \end{bmatrix}, \tag{6}$$

and $\boldsymbol{\varepsilon}(t) \triangleq [s(t), e(t)]^{\mathrm{T}} \in \mathbb{R}^2$ consists of the *along-track distance* $s(t)$ and the *cross-track error* $e(t)$, see Fig. 5. For path-following purposes, only the cross-track error is relevant since

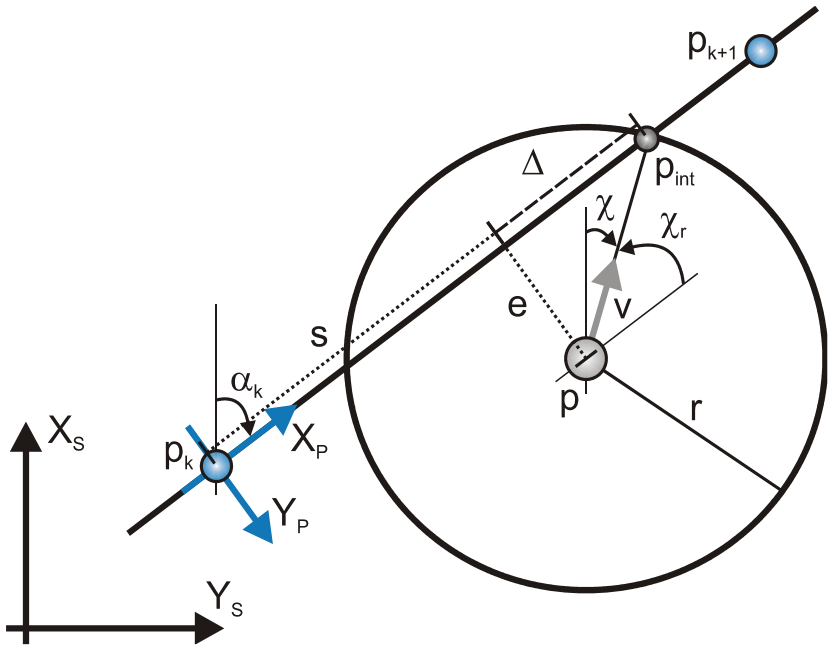


Fig. 5. The main variables associated with steering laws for straight-line paths

$e(t) = 0$ means that the vehicle has converged to the straight line. Expanding (5), the cross-track error can be explicitly stated by

$$e(t) = -(x(t) - x_k)\sin\alpha_k + (y(t) - y_k)\cos\alpha_k, \tag{7}$$

and the associated control objective for straight-line path following becomes

$$\lim_{t\to\infty} e(t) = 0. \tag{8}$$

In the following, two steering laws that ensure stabilization of $e(t)$ to the origin will be presented. The first method is used in ship motion control systems (Fossen 2002), and will be referred to as enclosure-based steering. The second method is called lookahead-based steering, and has links to the classical guidance principles from the missile literature. The two steering methods essentially operate by the same principle, but as will be made clear, the lookahead-based scheme has several advantages over the enclosure-based approach.

4.1.1 Enclosure-based steering

Imagine a circle with radius $r > 0$ enclosing $\mathbf{p}(t)$. If the circle radius is chosen sufficiently large, the circle will intersect the straight line at two points. The enclosure-based strategy for driving $e(t)$ to zero is then to direct the velocity toward the intersection point that corresponds to the desired direction of travel, which is implicitly defined by the sequence in which the waypoints are ordered. Such a solution involves directly assigning

$$\chi(t) = \text{atan2}(y_{\text{int}}(t) - y(t), x_{\text{int}}(t) - x(t)), \tag{9}$$

where $\mathbf{p}_{\text{int}}(t) \triangleq \left[x_{\text{int}}(t), y_{\text{int}}(t)\right]^{\text{T}} \in \mathbb{R}^2$ represents the intersection point of interest. In order to calculate $\mathbf{p}_{\text{int}}(t)$ (two unknowns), the following two equations must be solved

$$(x_{\text{int}}(t) - x(t))^2 + (y_{\text{int}}(t) - y(t))^2 = r^2 \tag{10}$$

$$\tan(\alpha_{\text{k}}) = \frac{y_{\text{k+1}} - y_{\text{k}}}{x_{\text{k+1}} - x_{\text{k}}}$$

$$= \frac{y_{\text{int}}(t) - y_{\text{k}}}{x_{\text{int}}(t) - x_{\text{k}}}, \tag{11}$$

where (10) represents the theorem of Pythagoras, while (11) states that the slope of the line between the two waypoints is constant. These equations are solved in the following, temporarily dropping the time dependence of the variables for notational convenience.
Denote the difference between the x- and y-position of the two waypoints as $\Delta x \triangleq x_{\text{k+1}} - x_{\text{k}}$ and $\Delta y \triangleq y_{\text{k+1}} - y_{\text{k}}$, respectively. The equations are first solved analytically assuming that $|\Delta x| > 0$ and secondly for the case $\Delta x = 0$.

Case 1: $|\Delta x| > 0$

Equation (11) results in

$$y_{\text{int}} = \left(\frac{\Delta y}{\Delta x}\right)(x_{\text{int}} - x_{\text{k}}) + y_{\text{k}} \tag{12}$$

when choosing to solve for y_{int}. For simplicity and brevity in the calculations to follow, denote

$$d \triangleq \left(\frac{\Delta y}{\Delta x}\right)$$

$$e \triangleq x_{\text{k}}$$

$$f \triangleq y_{\text{k}}.$$

Writing out (10), yields

$$x_{\text{int}}^2 - 2xx_{\text{int}} + x^2 + y_{\text{int}}^2 - 2yy_{\text{int}} + y^2 = r^2, \tag{13}$$

where

$$y_{\text{int}}^2 = \left(\left(\frac{\Delta y}{\Delta x}\right)(x_{\text{int}} - x_{\text{k}}) + y_{\text{k}}\right)^2$$

$$= (dx_{\text{int}} + (f - de))^2$$

$$= (dx_{\text{int}} + g)^2$$

$$= d^2x_{\text{int}}^2 + 2dgx_{\text{int}} + g^2, \tag{14}$$

where

$$g \triangleq f - de = y_k - \left(\frac{\Delta y}{\Delta x}\right) x_k$$

has been used. Subsequently, consider

$$2yy_{int} = 2y(dx_{int} + g) = 2dyx_{int} + 2gy, \tag{15}$$

such that (14) and (15) inserted into (13) gives

$$(1 + d^2)x_{int}^2 + 2(dg - dy - x)x_{int} + (x^2 + y^2 + g^2 - 2gy - r^2) = 0, \tag{16}$$

which is a standard, analytically-solvable second order equation. Then, denote

$$a \triangleq 1 + d^2$$

$$b \triangleq 2(dg - dy - x)$$

$$c \triangleq x^2 + y^2 + g^2 - 2gy - r^2,$$

from which the solution of (16) becomes

$$x_{int} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}, \tag{17}$$

where if $\Delta x > 0$, then $x_{int} = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$, and if $\Delta x < 0$, then $x_{int} = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$. Having calculated x_{int}, y_{int} is easily obtained from (12). Note that when $\Delta y = 0$, $y_{int} = y_k$ $(= y_{k+1})$.

Case 2: $\Delta x = 0$

If $\Delta x = 0$, only equation (10) is valid, which means that

$$y_{int} = y \pm \sqrt{r^2 - (x_{int} - x)^2}, \tag{18}$$

where $x_{int} = x_k$ $(= x_{k+1})$. If $\Delta y > 0$, then $y_{int} = y + \sqrt{r^2 - (x_{int} - x)^2}$, and if $\Delta y < 0$, then $y_{int} = y - \sqrt{r^2 - (x_{int} - x)^2}$. When $\Delta x = 0$, $\Delta y = 0$ is not an option.

4.1.2 Lookahead-based steering

Here, the steering assignment is separated into two parts

$$\chi(e) = \chi_p + \chi_r(e), \tag{19}$$

where

$$\chi_p = \alpha_k \tag{20}$$

is the path-tangential angle, while

$$\chi_{\mathrm{r}}(e) \triangleq \arctan\left(-\frac{e(t)}{\Delta}\right) \tag{21}$$

is a velocity-path relative angle which ensures that the velocity is directed toward a point on the path that is located a *lookahead distance* $\Delta > 0$ ahead of the direct projection of $\mathbf{p}(t)$ onto the path (Papoulias 1991), see Fig. 5.

As can be immediately noticed, this lookahead-based steering scheme is less computationally intensive than the enclosure-based approach. It is also valid for all cross-track errors, whereas the enclosure-based strategy requires $r \geq |e(t)|$. Furthermore, Fig. 5 shows that

$$e^2 + \Delta^2 = r^2, \tag{22}$$

which means that the enclosure-based approach corresponds to a lookahead-based scheme with a time-varying $\Delta(t) = \sqrt{r^2 - e(t)^2}$, varying between 0 (when $|e(t)| = r$) and r (when $|e(t)| = 0$). Only lookahead-based steering will be considered in the following.

4.2 Piecewise linear paths

If a path is made up of n straight-line segments connected by n+1 waypoints, a strategy must be employed to purposefully switch between these segments as they are traversed. In (Fossen 2002), it is suggested to associate a so-called circle of acceptance with each waypoint, with radius $R_{\mathrm{k+1}} > 0$ for waypoint $k+1$, such that the corresponding switching criterion becomes

$$(x_{\mathrm{k+1}} - x(t))^2 + (y_{\mathrm{k+1}} - y(t))^2 \leq R_{\mathrm{k+1}}^2, \tag{23}$$

i.e., to switch when $\mathbf{p}(t)$ has entered the waypoint-enclosing circle. Note that for the enclosure-based approach, such a switching criterion entails the additional (conservative) requirement $r \geq R_{\mathrm{k+1}}$.

A perhaps more suitable switching criterion solely involves the along-track distance $s(t)$, such that if the total along-track distance between waypoints $\mathbf{p}_{\mathrm{k}}$ and $\mathbf{p}_{\mathrm{k+1}}$ is denoted $s_{\mathrm{k+1}}$, a switch is made when

$$(s_{\mathrm{k+1}} - s(t)) \leq R_{\mathrm{k+1}}, \tag{24}$$

which is similar to (23), but has the advantage that $\mathbf{p}(t)$ does not need to enter the waypoint-enclosing circle for a switch to occur, i.e., no restrictions are put on the cross-track error. Thus, if no intrinsic value is associated with visiting the waypoints, and their only purpose is to implicitly define a piecewise linear path, there is no reason to apply the circle-of-acceptance switching criterion (23).

4.3 Steering for circles

Denote the center of a circle with radius $r_{\mathrm{c}} > 0$ as $\mathbf{p}_{\mathrm{c}} \triangleq [x_{\mathrm{c}}, y_{\mathrm{c}}]^{\mathrm{T}} \in \mathbb{R}^2$. Subsequently, consider a path-fixed reference frame with origin at the direct projection of $\mathbf{p}(t)$ onto the circular

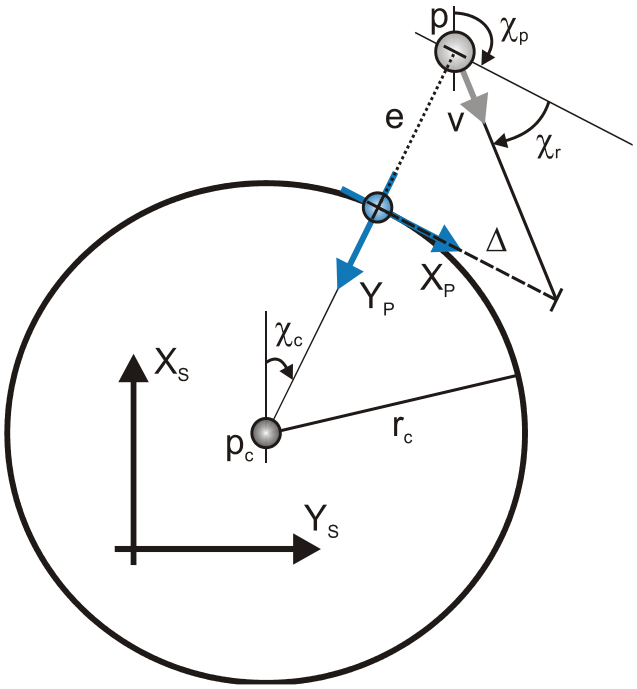


Fig. 6. The main variables associated with steering for circles

path, see Fig. 6. The x-axis of this reference frame has been rotated a positive angle (relative to the x-axis of the stationary reference frame)

$$\chi_p(t) = \chi_c(t) + \lambda\frac{\pi}{2}, \tag{25}$$

where

$$\chi_c(t) \triangleq \text{atan2}\big(y(t) - y_c, x(t) - x_c\big), \tag{26}$$

and $\lambda \in \{-1,1\}$ with $\lambda = -1$ corresponding to anti-clockwise motion and $\lambda = 1$ to clockwise motion. Hence, χ_p becomes time-varying for circular (curved) motion, as opposed to the constant χ_p associated with straight lines (20). Also, note that (26) is undefined for $\mathbf{p}(t) = \mathbf{p}_c$, i.e., when the kinematic vehicle is located at the circle center. In this case, any projection of $\mathbf{p}(t)$ onto the circular path is valid, but in practice this problem can be alleviated by, e.g., purposefully choosing $\chi_c(t)$ based on the motion of $\mathbf{p}(t)$.

Since the path-following control objective for circles is identical to (8), lookahead-based steering can be employed, implemented by using (19) with (25) instead of (20), and

$$\begin{aligned} e(t) &= r_c - \big|\mathbf{p}(t) - \mathbf{p}_c\big| \\ &= r_c - \sqrt{(x(t) - x_c)^2 - (y(t) - y_c)^2} \end{aligned} \tag{27}$$

in (21), see Fig. 6. Note that the lookahead distance Δ is no longer defined along the path, but (in general) along the x-axis of the path-fixed frame (i.e., along the path tangential associated with the origin of the path-fixed frame). An along-track distance $s(t)$ can also be computed relative to some fixed point on the circle perimeter if required.

4.4 Steering for regularly parameterized paths

Consider a planar path continuously parameterized by a scalar variable $\varpi \in \mathbb{R}$, such that the position of a point belonging to the path is represented by $\mathbf{p}_{\mathrm{p}}(\varpi) \in \mathbb{R}^2$. Thus, the path is a one-dimensional manifold that can be expressed by the set

$$\mathcal{P} \triangleq \left\{ \mathbf{p} \in \mathbb{R}^2 \mid \mathbf{p} = \mathbf{p}_{\mathrm{p}}(\varpi) \ \forall \ \varpi \in \mathbb{R} \right\}. \tag{28}$$

Regularly parameterized paths belong to the subset of $\mathcal{P}$ for which $\left| \mathbf{p}_{\mathrm{p}}'(\varpi) \right| \triangleq \left| \mathrm{d}\mathbf{p}_{\mathrm{p}}(\varpi) / \mathrm{d}\varpi \right|$ is non-zero and finite, which means that such paths never degenerate into a point nor have corners. These paths include both straight lines (zero curvature) and circles (constant curvature). However, most are paths with varying curvature. For such paths, it is not trivial to calculate the cross-track error $e(t)$ required in (21).

Although it is possible to calculate the exact projection of $\mathbf{p}(t)$ onto the path by applying the so-called Serret-Frenet equations, such an approach suffers from a kinematic singularity associated with the osculating circle of the instantaneous projection point (Samson 1992). For every point along a curved path, there exists an associated tangent circle with radius $r(\varpi) = 1 / c(\varpi)$, where $c(\varpi)$ is the curvature at the path point. This circle is known as the osculating circle, and if at any time $\mathbf{p}(t)$ is located at the origin of the osculating circle, the projected point on the path will have to move infinitely fast, which is not possible. This kinematic singularity effect necessitates a different approach to obtain the cross-track error required for steering purposes. The solution considered here seems to first have been suggested in (Aicardi et al. 1995), then refined and put into a differential-geometric framework in (Lapierre et al. 2003), and finally extended into the form presented below in (Breivik & Fossen 2004b).

Thus, consider an arbitrary path point $\mathbf{p}_{\mathrm{p}}(\varpi)$. Subsequently, consider a path-fixed reference frame with origin at $\mathbf{p}_{\mathrm{p}}(\varpi)$, whose x-axis has been rotated a positive angle (relative to the x-axis of the stationary reference frame)

$$\chi_{\mathrm{p}}(\varpi) = \operatorname{atan2}\left(y_{\mathrm{p}}'(\varpi), x_{\mathrm{p}}'(\varpi) \right), \tag{29}$$

such that

$$\boldsymbol{\varepsilon}(t) = \mathbf{R}(\chi_{\mathrm{p}})^{\mathrm{T}} (\mathbf{p}(t) - \mathbf{p}_{\mathrm{p}}(\varpi)), \tag{30}$$

where $\boldsymbol{\varepsilon}(t) = \left[s(t), e(t) \right]^{\mathrm{T}} \in \mathbb{R}^2$ represents the *along-track* and *cross-track errors* relative to $\mathbf{p}_{\mathrm{p}}(\varpi)$, decomposed in the path-fixed reference frame by

$$\mathbf{R}(\chi_{\mathrm{p}}) = \begin{bmatrix} \cos\chi_{\mathrm{p}} & -\sin\chi_{\mathrm{p}} \\ \sin\chi_{\mathrm{p}} & \cos\chi_{\mathrm{p}} \end{bmatrix}. \tag{31}$$

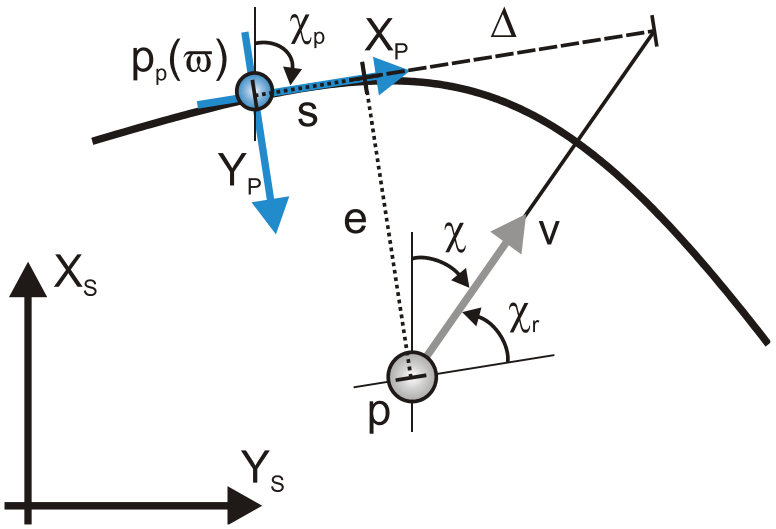


Fig. 7. The main variables associated with steering for regularly parameterized paths

In contrast to (8), the path-following control objective now becomes

$$\lim_{t\to\infty} \boldsymbol{\varepsilon}(t) = \mathbf{0}, \tag{32}$$

and in order to reduce $\boldsymbol{\varepsilon}(t)$ to zero, $\mathbf{p}(t)$ and $\mathbf{p}_p(\varpi)$ can collaborate with each other. Specifically, $\mathbf{p}_p(\varpi)$ can contribute by moving toward the direct projection of $\mathbf{p}(t)$ onto the x-axis of the path-fixed reference frame by assigning

$$\dot{\varpi} = \frac{U(t)\cos\chi_r(e) + \gamma s(t)}{\left|\mathbf{p}_p'(\varpi)\right|}, \tag{33}$$

where $\chi_r(e)$ is given by (21), $\gamma > 0$, and $\left|\mathbf{p}_p'(\varpi)\right| = \sqrt{x_p'(\varpi)^2 + y_p'(\varpi)^2}$. As can be seen, the first element of the numerator represents a kinematic feedforward of the projected speed of $\mathbf{p}(t)$ onto the path tangential, while the second element represents a linear feedback term whose purpose is to reduce the along-track error to zero. Hence, the path-constrained attractor $\mathbf{p}_p(\varpi)$ tracks the motion of $\mathbf{p}(t)$, which steers by the location of $\mathbf{p}_p(\varpi)$ through the cross-track error of (30) by employing (19) with (29) and (21) for $U(t) > 0$. Such an approach suffers from no kinematic singularities, and ensures that $\boldsymbol{\varepsilon}(t)$ is reduced to zero for regularly parameterized paths. To avoid initial transients in $e(t)$, the initial along-track error $s(0)$ can be minimized offline.

4.4.1 Relations to classical guidance laws

Drawing a connection to the classical guidance principles of the missile literature, lookahead-based steering can be interpreted as pure pursuit of the lookahead point. Convergence to $\mathbf{p}_p(\varpi)$ is thus achieved as $\mathbf{p}(t)$ in vain chases a carrot located a distance Δ further ahead along the path tangential. However, in (Papoulias 1992), the lookahead point is suggested to be placed further ahead along the path instead of along the path tangential,

which leads to a steady-state offset in the cross-track error for curved paths. In this case, the velocity of $\mathbf{p}(t)$ cannot be aligned with the velocity of $\mathbf{p}_{\mathrm{p}}(\varpi)$ for zero cross-track error. This distinction is vital for curved paths, but not for straight-line paths, where the path tangential is always directed along the path. Thus, in general, the pursued carrot must be located along the path tangential and not along the path itself. Nevertheless, the along-path approach has been widely reported in the literature, see, e.g., (Ollero & Heredia 1995), (Rankin et al. 1997), and (Castaño et al. 2005).

4.4.2 Off-path traversing of curved paths

In some applications, it can be desirable to perform off-path traversing of regularly parameterized paths. Specifically, off-path traversing of curved paths requires the use of two virtual points to avoid kinematic singularities. This concept was originally suggested in (Breivik et al. 2006), and used for formation control of ships in (Breivik et al. 2008).

4.4.3 Path parameterizations

Although the recently-presented guidance method also can be applied for both straight lines and circles, the analytic, path-specific approaches presented previously are often preferable since they do not require numerical integrations such as (33). However, for completeness, applicable (arc-length) parameterizations of straight lines and circles are given in the following.

Parameterization of straight lines

A planar straight line can be parameterized by $\varpi \in \mathbb{R}$ as

$$x_{\mathrm{p}}(\varpi) = x_{\mathrm{f}} + \varpi \cos\alpha \tag{34}$$

$$y_{\mathrm{p}}(\varpi) = y_{\mathrm{f}} + \varpi \sin\alpha, \tag{35}$$

where $\mathbf{p}_{\mathrm{f}} \triangleq [x_{\mathrm{f}}, y_{\mathrm{f}}]^{\mathrm{T}} \in \mathbb{R}^2$ represents a fixed point on the path (for which ϖ is defined relative to), and $\alpha \in \mathbb{S}$ represents the orientation of the path relative to the x-axis of the stationary reference frame (corresponding to the direction of increasing ϖ).

Parameterization of circles

A planar circle can be parameterized by $\varpi \in \mathbb{R}$ as

$$x_{\mathrm{p}}(\varpi) = x_{\mathrm{c}} + r_{\mathrm{c}} \cos\left(\frac{\varpi}{r_{\mathrm{c}}}\right) \tag{36}$$

$$y_{\mathrm{p}}(\varpi) = y_{\mathrm{c}} + \lambda r_{\mathrm{c}} \sin\left(\frac{\varpi}{r_{\mathrm{c}}}\right), \tag{37}$$

where $\mathbf{p}_{\mathrm{c}} = [x_{\mathrm{c}}, y_{\mathrm{c}}]^{\mathrm{T}} \in \mathbb{R}^2$ represents the circle center, $r_{\mathrm{c}} > 0$ represents the circle radius, and $\lambda \in \{-1, 1\}$ decides in which direction $\mathbf{p}_{\mathrm{p}}(\varpi)$ traces the circumference; $\lambda = -1$ for anti-clockwise motion and $\lambda = 1$ for clockwise motion.

4.5 Speed law for path tracking

As previously stated, the control objective of a path-tracking scenario is to track a target that is constrained to move along a path. Denoting the path-parameterization variable associated with the path-traversing target by $\varpi_t(t) \in \mathbb{R}$, the control objective is identical to (1) with $\mathbf{p}_t(t) = \mathbf{p}_p(\varpi_t(t))$. Here, $\varpi_t(t)$ can be updated by

$$\dot{\varpi}_t = \frac{U_t(t)}{\left|\mathbf{p}_p'(\varpi_t)\right|}, \tag{38}$$

which means that the target point traverses the path with the speed profile $U_t(t) > 0$, which can also be made to vary with ϖ_t.

Naturally, this problem can be solved by the target-tracking methods of Section 3, e.g., through the direct velocity assignment (2). However, by using such methods, all available path information is disregarded, and $\mathbf{p}(t)$ will appear to be "cutting corners" in its pursuit of $\mathbf{p}_p(\varpi_t(t))$, seeing only $\mathbf{p}_t(t)$.

Another approach is to employ the path knowledge that is apriori available, to divide the path-tracking problem into two tasks, i.e., a spatial task and a temporal task (Skjetne et al. 2004). The spatial task was just solved in the previous part, while the temporal task can be solved by employing the speed law

$$U(t) = \left|\mathbf{p}_p'(\varpi)\right| \left(\frac{U_t(t)}{\left|\mathbf{p}_p'(\varpi_t)\right|} - \mu \frac{\tilde{\varpi}(t)}{\sqrt{\tilde{\varpi}(t)^2 + \Delta_{\tilde{\varpi}}^2}} \right), \tag{39}$$

where

$$\tilde{\varpi}(t) \triangleq \varpi(t) - \varpi_t(t), \tag{40}$$

μ can be chosen as

$$\mu = \rho \frac{U_t(t)}{\left|\mathbf{p}_p'(\varpi_t)\right|}, \ \rho \in \langle 0, 1], \tag{41}$$

and where $\Delta_{\tilde{\varpi}} > 0$ specifies the rendezvous behavior toward the target, such that

$$U(t) = U_t(t) \left(1 - \rho \frac{\tilde{\varpi}(t)}{\sqrt{\tilde{\varpi}(t)^2 + \Delta_{\tilde{\varpi}}^2}} \right) \frac{\left|\mathbf{p}_p'(\varpi(t))\right|}{\left|\mathbf{p}_p'(\varpi_t(t))\right|}, \tag{42}$$

which means that the kinematic vehicle speeds up to catch the target when located behind it, and speeds down to wait when located in front of it. Hence, this approach entails a synchronization-law extension of the path-following scenario, where no corners are cut.

4.6 Path maneuvering aspects

The path-maneuvering scenario involves the use of knowledge about vehicle maneuverability constraints to design purposeful speed and steering laws that allow for feasible path negotiation. Since this work only deals with kinematic considerations, such

deliberations are outside of its scope. However, relevant work in this vein include (Sheridan 1966), (Yoshimoto et al. 2000), (Skjetne et al. 2004), (Børhaug et al. 2006), (Subbotin et al. 2006), (Gomes et al. 2006), and (Sharp 2007). Much work still remains to be done on this topic, which represents a rich source of interesting and challenging problems.

4.7 Steering laws as saturated control laws

Rewriting (21) as

$$\chi_{\mathrm{r}}(e) = \arctan\left(-k_{\mathrm{p}} e(t)\right),\ k_{\mathrm{p}} = \frac{1}{\Delta} > 0, \tag{43}$$

it can be seen that the lookahead-based steering law is equivalent to a saturated proportional control law, effectively mapping $e \in \mathbb{R}$ into $\chi_{\mathrm{r}}(e) \in \langle -\pi/2, \pi/2 \rangle$.

As can be inferred from the geometry of Fig. 5, a small lookahead distance implies aggressive steering, which intuitively is confirmed by a correspondingly large proportional gain in the saturated control interpretation. This interpretation also suggests the possibility of introducing, e.g., integral action into the steering law, such that

$$\chi_{\mathrm{r}}(e) = \arctan\left(-k_{\mathrm{p}} e(t) - k_{\mathrm{i}} \int_0^t e(\tau)\mathrm{d}\tau\right), \tag{44}$$

where $k_{\mathrm{i}} > 0$ represents the integral gain. Note that such integral action is not necessary in a purely kinematic setting, but can be particularly useful for underactuated AUVs that can only steer by attitude information, enabling them to follow straight-line paths while under the influence of constant ocean currents even without having access to velocity information. Thus, considering horizontal path following along straight lines, the desired yaw angle can be computed by

$$\psi_{\mathrm{d}}(e) = \alpha_{\mathrm{k}} + \chi_{\mathrm{r}}(e) \tag{45}$$

with $\chi_{\mathrm{r}}(e)$ as in (44). In practice, to avoid overshoot and windup effects, care must be taken when using integral action in the steering law. Specifically, the integral term should only be used when a steady-state off-track condition has been detected.

For those AUVs that do have access to velocity information, temporal integration can be replaced by spatial integration in order to minimize overshoot and windup problems (Davidson et al. 2002), employing

$$\chi_{\mathrm{r}}(e) = \arctan\left(-k_{\mathrm{p}} e(t) - k_{\mathrm{i}} \int_0^s e(\sigma)\mathrm{d}\sigma\right), \tag{46}$$

where for straight-line paths

$$\int_0^s e(\sigma)\mathrm{d}\sigma = \int_0^t e(\tau)\frac{\mathrm{d}\sigma}{\mathrm{d}\tau}\mathrm{d}\tau \tag{47}$$

$$= \int_0^t e(\tau) U(\tau) \cos(\chi(\tau) - \alpha_{\mathrm{k}})\mathrm{d}\tau, \tag{48}$$

which means that integration only occurs when the velocity has a component along the path. Also, derivative action can be added to the steering law in order to obtain a damped transient response toward the path.

5. Guidance laws for 3D scenarios

In this section, guidance laws for 3D motion control scenarios are considered. For spatial target-tracking purposes, the guidance principles of Section 3 remain equally valid, and the velocity assignment (2) is directly applicable for 3D target tracking. However, the steering laws of Section 4 need to be extended. Specifically, in what follows, lookahead-based steering will be put into a spatial framework for regularly parameterized paths, adapted from (Breivik & Fossen 2005b). Note that the path-tracking speed law (42) need not be modified, and can be directly applied to 3D scenarios.

Now, represent the kinematic vehicle by its spatial position $\mathbf{p}(t) \triangleq \left[x(t), y(t), z(t)\right]^{\mathrm{T}} \in \mathbb{R}^3$ and velocity $\mathbf{v}(t) \triangleq \dot{\mathbf{p}}(t) \in \mathbb{R}^3$, stated relative to some stationary reference frame. Also, the speed is represented by $U(t) \triangleq \left|\mathbf{v}(t)\right| = \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2 + \dot{z}(t)^2} \geq 0$, while the steering is characterized by the two angular variables $\chi(t) \triangleq \operatorname{atan2}\left(\dot{y}(t), \dot{x}(t)\right) \in \mathbb{S}$ (the azimuth angle) and $\upsilon(t) \triangleq \operatorname{atan2}\left(-\dot{z}(t), \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}\right) \in \mathbb{S}$ (the elevation angle). Path following is then ensured by proper assignments to $\chi(t)$ and $\upsilon(t)$ as long as $U(t) > 0$.

Then, consider a spatial path continuously parameterized by a scalar variable $\varpi \in \mathbb{R}$, such that the position of a point belonging to the path is represented by $\mathbf{p}_{\mathrm{p}}(\varpi) \in \mathbb{R}^3$. Thus, the path can be expressed by the set

$$\mathcal{P} \triangleq \left\{ \mathbf{p} \in \mathbb{R}^3 \mid \mathbf{p} = \mathbf{p}_{\mathrm{p}}(\varpi) \ \forall \ \varpi \in \mathbb{R} \right\}. \tag{49}$$

Subsequently, consider an arbitrary path point $\mathbf{p}_{\mathrm{p}}(\varpi)$, and define a path-fixed reference frame with origin at this point. Starting with the same orientation as the stationary frame, two consecutive elementary rotations can be performed to arrive at this path-fixed frame. The first is to positively rotate the stationary frame an angle

$$\chi_{\mathrm{p}}(\varpi) = \operatorname{atan2}\left(y_{\mathrm{p}}'(\varpi), x_{\mathrm{p}}'(\varpi)\right) \tag{50}$$

about its z-axis, while the second is to positively rotate the resulting intermediate frame an angle

$$\upsilon_{\mathrm{p}}(\varpi) = \operatorname{atan2}\left(-z_{\mathrm{p}}'(\varpi), \sqrt{x_{\mathrm{p}}'(\varpi)^2 + y_{\mathrm{p}}'(\varpi)^2}\right) \tag{51}$$

about its y-axis. These rotations can also be represented by the rotation matrices

$$\mathbf{R}(\chi_{\mathrm{p}}) \triangleq \begin{bmatrix} \cos\chi_{\mathrm{p}} & -\sin\chi_{\mathrm{p}} & 0 \\ \sin\chi_{\mathrm{p}} & \cos\chi_{\mathrm{p}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{52}$$

and

$$\mathbf{R}(\upsilon_{\mathrm{p}}) \triangleq \begin{bmatrix} \cos\upsilon_{\mathrm{p}} & 0 & \sin\upsilon_{\mathrm{p}} \\ 0 & 1 & 0 \\ -\sin\upsilon_{\mathrm{p}} & 0 & \cos\upsilon_{\mathrm{p}} \end{bmatrix}, \tag{53}$$

respectively. Hence, the full rotation can be represented by

$$\mathbf{R}(\chi_{\mathrm{p}}, \upsilon_{\mathrm{p}}) \triangleq \mathbf{R}(\chi_{\mathrm{p}})\mathbf{R}(\upsilon_{\mathrm{p}}), \tag{54}$$

such that

$$\boldsymbol{\varepsilon}(t) = \mathbf{R}(\chi_{\mathrm{p}}, \upsilon_{\mathrm{p}})^{\mathrm{T}}(\mathbf{p}(t) - \mathbf{p}_{\mathrm{p}}(\varpi)), \tag{55}$$

where $\boldsymbol{\varepsilon}(t) = \left[s(t), e(t), h(t)\right]^{\mathrm{T}} \in \mathbb{R}^3$ represents the *along-track*, *cross-track*, and *vertical-track errors* relative to $\mathbf{p}_{\mathrm{p}}(\varpi)$, decomposed in the path-fixed reference frame. The path-following control objective is identical to (32), and $\boldsymbol{\varepsilon}(t)$ can be reduced to zero by assigning an appropriate steering law to the velocity of $\mathbf{p}(t)$ as well as a purposeful collaborative behavior to $\mathbf{p}_{\mathrm{p}}(\varpi)$.

Specifically, the steering law involves

$$\chi_{\mathrm{r}}(e) \triangleq \arctan\left(-\frac{e(t)}{\Delta}\right), \tag{56}$$

which is equivalent to (21) with $\Delta > 0$, used to shape the convergence behavior toward the xz-plane of the path-fixed frame, and

$$\upsilon_{\mathrm{r}}(h) \triangleq \arctan\left(\frac{h(t)}{\sqrt{e(t)^2 + \Delta^2}}\right), \tag{57}$$

used to shape the convergence behavior toward the xy-plane of the path-fixed frame, see Fig. 8. Also, $\mathbf{p}_{\mathrm{p}}(\varpi)$ moves collaboratively toward the direct projection of $\mathbf{p}(t)$ onto the x-axis of the path-fixed reference frame by

$$\dot{\varpi} = \frac{U(t)\cos\chi_{\mathrm{r}}(e)\cos\upsilon_{\mathrm{r}}(h) + \gamma s(t)}{\left|\mathbf{p}_{\mathrm{p}}'(\varpi)\right|}, \tag{58}$$

where $\gamma > 0$ and $\left|\mathbf{p}_{\mathrm{p}}'(\varpi)\right| = \sqrt{x_{\mathrm{p}}'(\varpi)^2 + y_{\mathrm{p}}'(\varpi)^2 + z_{\mathrm{p}}'(\varpi)^2}$. In sum, four angular variables (50), (51), (56), and (57) are used to specify the 3D steering law required for path-following purposes. Fortunately, these variables can be compactly represented by the azimuth angle

$$\chi(\chi_{\mathrm{p}}, \upsilon_{\mathrm{p}}, \chi_{\mathrm{r}}, \upsilon_{\mathrm{r}}) = \operatorname{atan2}\left(f(\chi_{\mathrm{p}}, \upsilon_{\mathrm{p}}, \chi_{\mathrm{r}}, \upsilon_{\mathrm{r}}), g(\chi_{\mathrm{p}}, \upsilon_{\mathrm{p}}, \chi_{\mathrm{r}}, \upsilon_{\mathrm{r}})\right), \tag{59}$$

where

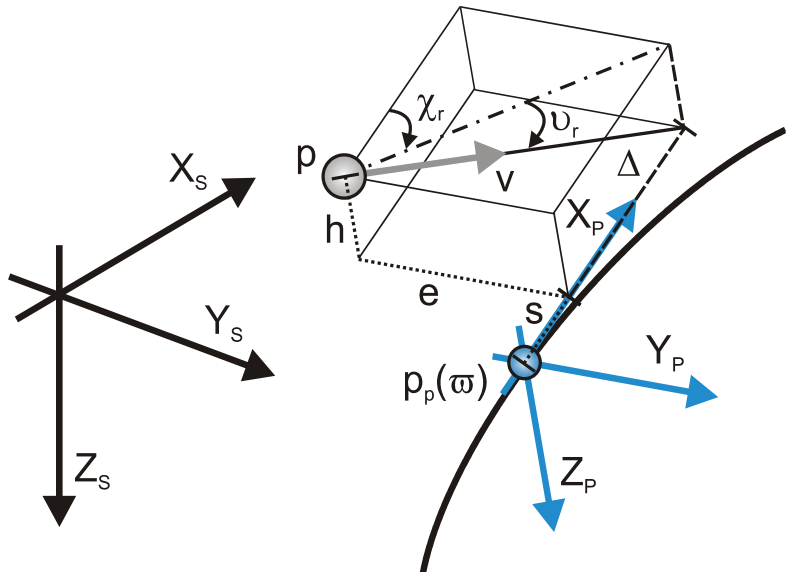


Fig. 8. The main variables associated with steering for regularly parameterized 3D paths

$$f(\chi_p,\upsilon_p,\chi_r,\upsilon_r) = \cos\chi_p \sin\chi_r \cos\upsilon_r - \sin\chi_p \sin\upsilon_p \sin\upsilon_r + \sin\chi_p \cos\upsilon_p \cos\chi_r \cos\upsilon_r \tag{60}$$

and

$$g(\chi_p,\upsilon_p,\chi_r,\upsilon_r) = -\sin\chi_p \sin\chi_r \cos\upsilon_r - \cos\chi_p \sin\upsilon_p \sin\upsilon_r + \cos\chi_p \cos\upsilon_p \cos\chi_r \cos\upsilon_r, \tag{61}$$

and the elevation angle

$$\upsilon(\upsilon_p,\chi_r,\upsilon_r) = \arcsin(\sin\upsilon_p \cos\chi_r \cos\upsilon_r + \cos\upsilon_p \sin\upsilon_r). \tag{62}$$

Through the use of trigonometric addition formulas, it can be shown that (59) is equivalent to (19) in the 2D case, i.e., when $\upsilon_p = \upsilon_r = 0$.

5.1 Path parameterizations

Applicable (arc-length) parameterizations of straight lines and helices are now given.

5.1.1 Parameterization of straight lines

A spatial straight line can be parameterized by $\varpi \in \mathbb{R}$ as

$$x_p(\varpi) = x_f + \varpi \cos\alpha \cos\beta \tag{63}$$

$$y_p(\varpi) = y_f + \varpi \sin\alpha \cos\beta \tag{64}$$

$$z_p(\varpi) = z_f - \varpi \sin\beta, \tag{65}$$

where $\mathbf{p}_\mathrm{f} \triangleq [x_\mathrm{f}, y_\mathrm{f}, z_\mathrm{f}]^\mathrm{T} \in \mathbb{R}^3$ represents a fixed point on the path (for which ϖ is defined relative to), and $\alpha \in \mathbb{S}$ represents the azimuth angle of the path, while $\beta \in \mathbb{S}$ represents the elevation angle of the path (both corresponding to the direction of increasing ϖ).

5.1.2 Parameterization of helices

A helix can be parameterized by $\varpi \in \mathbb{R}$ as

$$x_\mathrm{p}(\varpi) = x_\mathrm{c} + r_\mathrm{c} \cos\left(\frac{\varpi}{\sqrt{2} r_\mathrm{c}}\right) \tag{66}$$

$$y_\mathrm{p}(\varpi) = y_\mathrm{c} + \lambda r_\mathrm{c} \sin\left(\frac{\varpi}{\sqrt{2} r_\mathrm{c}}\right) \tag{67}$$

$$z_\mathrm{p}(\varpi) = z_\mathrm{c} - \frac{\varpi}{\sqrt{2}}, \tag{68}$$

where $\mathbf{p}_\mathrm{c} = [x_\mathrm{c}, y_\mathrm{c}, z_\mathrm{c}]^\mathrm{T} \in \mathbb{R}^3$ represents the origin of the helix center (for which ϖ is defined relative to), $r_\mathrm{c} > 0$ represents the radius of the horizontally-projected circle of the helix, and $\lambda \in \{-1, 1\}$ decides in which direction this horizontally-projected circle is traced; $\lambda = -1$ for anti-clockwise motion and $\lambda = 1$ for clockwise motion. Here, an increase in ϖ corresponds to movement in the negative direction of the z-axis of the stationary frame.

6. Conclusions

This work has given an overview of guidance laws applicable to motion control of AUVs in 2D and 3D. Specifically, considered scenarios have included target tracking, where only instantaneous information about the target motion is available, as well as path scenarios, where spatial information is available apriori. For target-tracking purposes, classical guidance laws from the missile literature were reviewed, in particular line of sight, pure pursuit, and constant bearing. For the path scenarios, enclosure-based and lookahead-based guidance laws were presented. Relations between the guidance laws have been discussed, as well as interpretations toward saturated control.

7. References

Aguiar, A. P. & Hespanha, J. P. (2004). Logic-based switching control for trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. In: *Proceedings of the ACC'04, Boston, Massachusetts, USA*

Aicardi, M.; Casalino, G.; Bicchi, A. & Balestrino, A. (1995). Closed loop steering of unicycle-like vehicles via Lyapunov techniques. *IEEE Robotics and Automation Magazine* 2(1), 27—35

Antonelli, G.; Fossen, T. I. & Yoerger, D. R. (2008). Underwater robotics. In: *Springer Handbook of Robotics* (B. Siciliano and O. Khatib, Eds.). pp. 987— 1008. Springer-Verlag Berlin Heidelberg

Battin, R. H. (1982). Space guidance evolution - A personal narrative. *Journal of Guidance, Control, and Dynamics* 5(2), 97—110

Blidberg, D. R. (2001). The development of autonomous underwater vehicles (AUVs); a brief summary. In: *Proceedings of the ICRA'01, Seoul, Korea*

Breivik, M. & Fossen, T. I. (2004a). Path following of straight lines and circles for marine surface vessels. In: *Proceedings of the 6th IFAC CAMS, Ancona, Italy*

Breivik, M. & Fossen, T. I. (2004b). Path following for marine surface vessels. In: *Proceedings of the OTO'04, Kobe, Japan*

Breivik, M. & Fossen, T. I. (2005a). Guidance-based path following for autonomous underwater vehicles. In: *Proceedings of the OCEANS'05, Washington D.C., USA*

Breivik, M. & Fossen, T. I. (2005b). Principles of guidance-based path following in 2D and 3D. In: *Proceedings of the CDC-ECC'05, Seville, Spain*

Breivik, M.; Subbotin, M. V. & Fossen, T. I. (2006). Kinematic aspects of guided formation control in 2D. In: *Group Coordination and Cooperative Control* (K. Y. Pettersen, J. T. Gravdahl and H. Nijmeijer, Eds.). pp. 54—74. Springer-Verlag Heidelberg

Breivik, M. & Fossen, T. I. (2007). Applying missile guidance concepts to motion control of marine craft. In: *Proceedings of the 7th IFAC CAMS, Bol, Croatia*

Breivik, M.; Hovstein, V. E. & Fossen, T. I. (2008). Ship formation control: A guided
leader-follower approach. In: *Proceedings of the 17th IFAC World Congress, Seoul, Korea*

Breivik, M. & Fossen, T. I. (2008). Guidance laws for planar motion control. In: *Proceedings of the CDC'08, Cancun, Mexico*

Børhaug, E. & Pettersen, K. Y. (2006). LOS path following for underactuated underwater vehicle. In: *Proceedings of the 7th IFAC MCMC, Lisbon, Portugal*

Børhaug, E.; Pettersen, K. Y. & Pavlov, A. (2006). An optimal guidance scheme for cross-track control of underactuated underwater vehicles. In: *Proceedings of the MED'06, Ancona, Italy*

Caccia, M.; Bruzzone, G. & Veruggio, G. (2000). Guidance of unmanned underwater vehicles: Experimental results. In: *Proceedings of the ICRA'00, San Francisco, California, USA*

Castaño, A. R.; Ollero, A.; Vinagre, B. M. & Chen, Y. Q. (2005). Synthesis of a spatial lookahead path tracking controller. In: *Proceedings of the 16th IFAC World Congress, Prague, Czech Republic*

Cloutier, J. R.; Evers, J. H. & Feeley, J. J. (1989). Assessment of air-to-air missile guidance and control technology. *IEEE Control Systems Magazine* 9(6), 27—34

Craven, P. J.; Sutton, R. & Burns, R. S. (1998). Control strategies for unmanned underwater vehicles. *Journal of Navigation* 51, 79—105

Davidson, M.; Bahl, V. & Moore, K. L. (2002). Spatial integration for a nonlinear path tracking control law. In: *Proceedings of the ACC'02, Anchorage, Alaska, USA*

Do, K. D. & Pan, J. (2003). Robust and adaptive path following for underactuated autonomous underwater vehicles. In: *Proceedings of the ACC'03, Denver, Colorado, USA*

Draper, C. S. (1971). Guidance is forever. *Navigation* 18(1), 26—50

Encarnação, P. & Pascoal, A. (2000). 3D path following for autonomous underwater vehicle. In: *Proceedings of the CDC'00, Sydney, Australia*

Fossen, T. I. (2002). *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics

Fossier, M. W. (1984). The development of radar homing missiles. *Journal of Guidance, Control, and Dynamics* 7(6), 641—651

Gomes, P.; Silvestre, C.; Pascoal, A. & Cunha, R. (2006). A path-following controller for the DELFIMx autonomous surface craft. In: *Proceedings of the 7th IFAC MCMC, Lisbon, Portugal*

Haeussermann, W. (1981). Developments in the field of automatic guidance and control of rockets. *Journal of Guidance and Control* 4(3), 225—239

Hagen, P. E.; Størkersen, N. J. & Vestgård, K. (2003). The HUGIN AUVs – Multi-role capability for challenging underwater survey operations. *EEZ International*

Healey, A. J. & Lienard, D. (1993). Multivariable sliding-mode control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering* 18(3), 327—339

Justh, E. W. & Krishnaprasad, P. S. (2006). Steering laws for motion camouflage. *Proceedings of the Royal Society A* 462(2076), 3629—3643

Lapierre, L.; Soetanto, D. & Pascoal, A. (2003). Nonlinear path following with applications to the control of autonomous underwater vehicles. In: *Proceedings of the CDC'03, Maui, Hawaii, USA*

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press

Lin, C.-F. (1991). *Modern Navigation, Guidance, and Control Processing, Volume II*. Prentice Hall, Inc.

Lin, C.-L. & Su, H.-W. (2000). Intelligent control theory in guidance and control system design: An overview. *Proceedings of the National Science Council, ROC* 24(1), 15—30

Locke, A. S. (1955). *Guidance*. D. Van Nostrand Company, Inc.

MacKenzie, D. A. (1990). *Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance*. MIT Press

Mizutani, A.; Chahl, J. S. & Srinivasan, M. V. (2003). Motion camouflage in dragonflies. *Nature* 423, 604

Ollero, A. & Heredia, G. (1995). Stability analysis of mobile robot path tracking. In: *Proceedings of the IROS'95, Pittsburgh, Pennsylvania, USA*

Papoulias, F. A. (1991). Bifurcation analysis of line of sight vehicle guidance using sliding modes. *International Journal of Bifurcation and Chaos* 1(4), 849—865

Papoulias, F. A. (1992). Guidance and control laws for vehicle pathkeeping along curved trajectories. *Applied Ocean Research* 14(5), 291—302

Pastrick, H. L.; Seltzer, S. M. & Warren, M. E. (1981). Guidance laws for short-range tactical missiles. *Journal of Guidance and Control* 4(2), 98—108

Piccardo, H. R. & Honderd, G. (1991). A new approach to on-line path planning and generation for robots in non-static environments. *Robotics and Autonomous Systems* 8(3), 187—201

Rankin, A. L.; Crane III, C. D. & Armstrong II, D. G. (1997). Evaluating a PID, pure pursuit, and weighted steering controller for an autonomous land vehicle. In: *Proceedings of the SPIE Mobile Robotics XII, Pittsburgh, Pennsylvania, USA*

Refsnes, J. E.; Sørensen, A. J. & Pettersen, K. Y. (2008). Model-based output feedback control of slender-body underactuated AUVs: Theory and experiments. *IEEE Transactions on Control Systems Technology* 16(5), 930—946

Roberts, G. N. & Sutton, R. (2006). *Advances in Unmanned Marine Vehicles*. The Institution of Electrical Engineers

Samson, C. (1992). Path following and time-varying feedback stabilization of a wheeled mobile robot. In: *Proceedings of the ICARCV'92, Singapore*

Sciavicco, L. & Siciliano, B. (2002). *Modelling and Control of Robot Manipulators*. Springer-Verlag London Ltd.

Sharp, R. S. (2007). Application of optimal preview control to speed-tracking of road vehicles. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 221(12), 1571—1578

Sheridan, T. B. (1966). Three models of preview control. *IEEE Transactions on Human Factors in Electronics* 7(2), 91—102

Shneydor, N. A. (1998). *Missile Guidance and Pursuit: Kinematics, Dynamics and Control*.
Horwood Publishing Ltd.

Siouris, G. M. (2004). *Missile Guidance and Control Systems*. Springer-Verlag New York, Inc.

Skjetne, R.; Fossen, T. I. & Kokotović, P. V. (2004). Robust output maneuvering for a class of nonlinear systems. *Automatica* 40(3), 373—383

Spearman, M. L. (1978). Historical development of worldwide guided missiles. In: *AIAA 16th Aerospace Sciences Meeting, Huntsville, Alabama, USA*

Subbotin, M. V.; Dăcić, D. B. & Smith, R. S. (2006). Preview based path-following in the presence of input constraints. In: *Proceedings of the ACC'06, Minneapolis, Minnesota, USA*

Valavanis, K. P.; Gracanin, D.; Matijasevic, M.; Kolluru, R. & Demetriou, G. A. (1997). Control architectures for autonomous underwater vehicles. *IEEE Control Systems Magazine* 17(6), 48—64

Wernli, R. L. (2000). AUV commercialization – Who's leading the pack?. In: *Proceedings of the OCEANS'00, Providence, Rhode Island, USA*

Westrum, R. (1999). *Sidewinder: Creative Missile Development at China Lake*. Naval Institute Press

Whitcomb, L. (2000). Underwater robotics: Out of the research laboratory and into the field. In: *Proceedings of the ICRA'00, San Francisco, California, USA*

White, B. A. & Tsourdos, A. (2001). Modern missile guidance design: An overview. In: *Proceedings of the IFAC Automatic Control in Aerospace, Bologna, Italy*

Yanushevsky, R. (2008). *Modern Missile Guidance*. CRC Press

Yoshimoto, K.; Katoh, M. & Inoue, K. (2000). A vision-based speed control algorithm for autonomous driving. In: *Proceedings of the AVEC'00, Ann Arbor, Michigan, USA*

Zarchan, P. (2002). *Tactical and Strategic Missile Guidance*. 4th ed.. American Institute of Aeronautics and Astronautics, Inc.

5

Enhanced Testing of Autonomous Underwater Vehicles using Augmented Reality & JavaBeans

Benjamin C. Davis and David M. Lane
Ocean Systems Laboratory, Heriot-Watt University
Scotland

1. Introduction

System integration and validation of embedded technologies has always been a challenge, particularly in the case of autonomous underwater vehicles (AUVs). The inaccessibility of the remote environment combined with the cost of field operations have been the main obstacles to the maturity and evolution of underwater technologies. Additionally, the analysis of embedded technologies is hampered by data processing and analysis time lags, due to low bandwidth data communications with the underwater platform. This makes real-world monitoring and testing challenging for the developer/operator as they are unable to react quickly or in real-time to the remote platform stimuli.

This chapter discusses the different testing techniques useful for unmanned underwater vehicle (UUVs) and gives example applications where necessary. Later sections digress into more detail about a new novel framework called the Augmented Reality Framework (ARF) and its applications on improving pre-real-world testing facilities for UUVs. To begin with more background is given on current testing techniques and their uses. To begin with some background is given about Autonomous Underwater Vehicles (AUVs).

An AUV (Healey et al., 1995) is a type of UUV. The difference between AUVs and Remotely operated vehicles (ROVs) is that AUVs employ intelligence, such as sensing and automatic decision making, allowing them to perform tasks autonomously, whilst ROVs are controlled remotely by a human with communications running down a tether. AUVs can operate for long periods of time without communication with an operator as they run a predefined mission plan. An operator can design missions for multiple AUVs and monitor their progress in parallel. ROVs require at least one pilot per ROV controlling them continuously. The cost of using AUVs should be drastically reduced compared with ROVs providing the AUV technology is mature enough to execute the task as well as an ROV. AUVs have no tether, or physical connection with surface vessels, and therefore are free to move without restriction around or inside complex structures. AUVs can be smaller and have lower powered thrusters than ROVs because they do not have to drag a tether behind them. Tethers can be thousands of metres in length for deep sea missions and consequently very heavy. In general, AUVs require less infrastructure than ROVs i.e. ROVs usually require a large ship and crew to operate which is not required with an AUV due to being easier to deploy and recover.

In general, autonomous vehicles (Zyda et al., 1990) can go where humans cannot, do not want to, or in more relaxed terms they are suited to doing the “the dull, the dirty, and the

dangerous". One of the main driving forces behind AUV development is automating , potentially tedious, tasks which take a long time to do manually and therefore incur large expenses. These can include oceanographic surveys, oil/gas pipeline inspection, cable inspection and clearing of underwater mine fields. These tasks can be monotonous for humans and can also require expensive ROV pilot skills. AUVs are well suited to labour intensive or repetitive tasks, and can perform their jobs faster and with higher accuracy than humans. The ability to venture into hostile or contaminated environments is something which makes AUVs particularly useful and cost efficient.

AUVs highlight a more specific problem. Underwater vehicles are expensive because they have to cope with the incredibly high pressures of the deepest oceans (the pressure increases by 1 atmosphere every 10m). The underwater environment itself is both hazardous and inaccessible which increases the costs of operations due to the necessary safety precautions. Therefore the cost of real-world testing, the later phase of the testing cycle, is particularly expensive in the case of UUVs. Couple this with poor communications with the remote platform (due to slow acoustic methods) and debugging becomes very difficult and time consuming. This incurs huge expenses, or more likely, places large constraints on the amount of real-world testing that can be feasibly done. It is paramount that for environments which are hazardous/inaccessible, such as sea, air and space, that large amounts of unnecessary real-world testing be avoided at all costs. Ideally, mixed reality testing facilities should be available for pre-real-world testing of the platform. However, due to the expense of creating specific virtual reality testing facilities themselves, adequate pre-real-world tests are not always carried out. This leads to failed projects crippled by costs, or worse, a system which is unreliable due to inadequate testing.

Different testing mechanisms can be used to keep real-world testing to a minimum. Hardware-in-the-loop (HIL), Hybrid Simulation (HS) and Pure Simulation (PS) are common pre-real-world testing methods. However, the testing harness created is usually very specific to the platform. This creates a problem when the user requires testing of multiple heterogeneous platforms in heterogeneous environments. Normally this requires many specific test harnesses, but creating them is often time consuming and expensive. Therefore, large amounts of integration tests are left until real-world trials, which is less than ideal.

Real world testing is not always feasible due to the high cost involved. It would be beneficial to test the systems in a laboratory first. One method of doing this is via pure simulation (PS) of data for each of the platform's systems. This is not a very realistic scenario as it doesn't test the actual system as a whole and only focuses on individual systems within a vehicle. The problem with PS alone is that system integration errors can go undetected until later stages of development, since this is when different modules will be tested working together. This can lead to problems later in the testing cycle by which time they are harder to detect and more costly to rectify. Therefore, as many tests as possible should to be done in a laboratory. A thorough testing cycle for a remote platform would include HIL, HS and PS testing scenarios. For example, an intuitive testing harness for HIL or HS would include: A 3D Virtual world with customisable geometry and terrain allowing for operator observation; A Sensor simulation suite providing exterioceptive sensor data which mimics the real world data interpreted by higher level systems; and a distributed communication protocol to allow for swapping of real for simulated systems running in different locations.

Thorough testing of the remote platform is usually left until later stages of development because creating a test harness for every platform can be complicated and costly. Therefore,

when considering a testing harness it is important that it is re-configurable and very generic in order to accommodate all required testing scenarios. The ability to extend the testing harness to use specialised modules is important so that it can be used to test specialized systems. Therefore a dynamic, extendible testing framework is required that allows the user to create modules in order to produce the testing scenario quickly and easily for their intended platform/environment.

2. Methods of testing

Milgrim's Reality-Virtuality continuum (Takemura et al., 1994), shown in Figure 1, depicts the continuum from reality to virtual reality and all the hybrid stages in between. The hybrid stages between real and virtual are known as augmented reality (Behringer et al., 2001) and augmented virtuality. The hybrid reality concepts are built upon by the ideas of Hardware-in-the-loop (HIL) and Hybrid Simulation (HS). Figure 1 shows how the different types of testing conform to the different types of mixed reality in the continuum. There are 4 different testing types:

1. **Pure Simulation** (PS) (Ridao et al., 2004) - testing of a platform's modules on an individual basis before being integrated onto the platform with other modules.
2. **Hardware-in-the-loop** (HIL) (Lane et al, 2001) - testing of the real integrated platform is carried out in a laboratory environment. Exterioceptive sensors such as sonar or video, which interact with the intended environment, may have to be simulated to fool the robot into thinking it is in the real world. This is very useful for integration testing as the entire system can be tested as a whole allowing for any system integration errors to be detected in advance of real world trials.
3. **Hybrid Simulation** (HS) (Ridao et al., 2004; Choi & Yuh, 2001) - testing the platform in its intended environment in conjunction with some simulated sensors driven from a virtual environment. For example, virtual objects can be added to the real world and the exterioceptive sensor data altered so that the robot thinks that something in the sensor dataset is real. This type of system is used if some higher level modules are not yet reliable enough to be trusted to behave as intended using real data. Consequently, fictitious data is used instead, augmented with the real data, and inputted to the higher level systems. Thus, if a mistake is made it doesn't damage the platform. An example of this is discussed in section 4.2.
4. **Real world testing** - This is the last stage of testing. When all systems are trusted the platform is ready for testing in the intended environment. All implementation errors should have been fixed in the previous stages otherwise this stage is very costly. For this stage to be as useful as possible the system designers and programmers need to have reliable intuitive feedback, in a virtual environment, about what the platform is doing otherwise problems can be very hard to see and diagnose.

ARF provides functionality across all stages of the continuum allowing for virtually any testing scenario to be realised. For this reason it is referred to as a mixed reality framework.

In the case of Augmented Reality, simulated data is added to the real world perception of some entity. For example, sonar data on an AUV could be altered so that it contains fictitious objects i.e. objects which are not present in the real world, but which are present in the virtual world. This can be used to test the higher level systems of an AUV such as obstacle detection (See *Obstacle detection and avoidance example in Section 4.2).* A virtual world is used to generate synthetic sensor data which is then mixed with the real world data. The

virtual world has to be kept in precise synchronization with the real world. This is commonly known in Augmented Reality as the *registration* problem. The accuracy of registration is dependent on the accuracy of the position/navigation systems onboard the platform. Registration is a well known problem with underwater vehicles when trying to match different sensor datasets to one another for visualisation. Accurate registration is paramount for displaying the virtual objects in the correct position in the simulated sensor data.

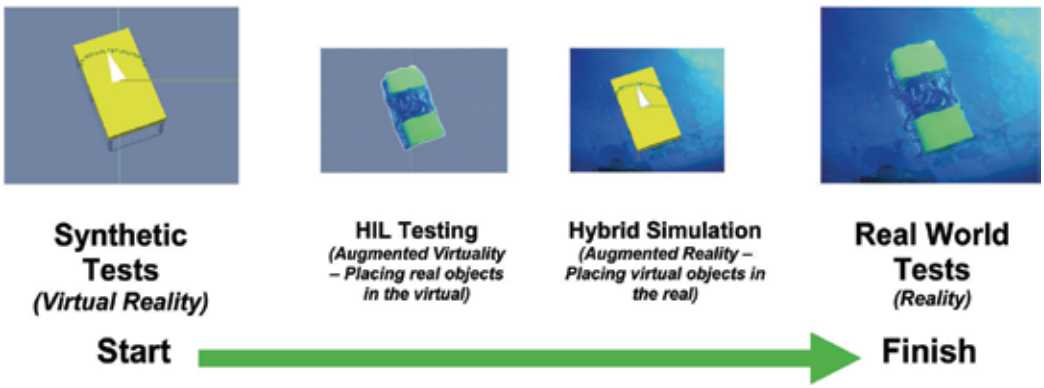


Fig. 1. Reality Continuum combined with Testing Types.

Augmented Virtuality is the opposite of augmented reality i.e. instead of being from a robot's/person's perspective it is from the virtual world's perspective - the virtual world is augmented with real world data. For example, real data collected by an AUV's sensors is rendered in real time in the virtual world in order to recreate the real world in virtual reality. This can be used for *Online Monitoring* (OM) and *operator training* (TR) (Ridao et al., 2004). This allows an AUV/ROV operator to see how the platform is situated in the remote environment, thus increasing situational awareness.

In Hybrid Simulation the platform operates in the real environment in conjunction with some sensors being simulated in real time by a synchronized virtual environment. Similar to Augmented Reality, the virtual environment is kept in synchronization using position data transmitted from the remote platform. Thus simulated sensors are attached to the virtual platform and moved around in synchronization with the real platform. Simulated sensors collect data from the virtual world and transmit the data back to the real systems on the remote platform. The real systems then interpret this data as if it were real. It is important that simulated data is very similar to the real data so that the higher level systems cannot distinguish between the two. In summary, the real platform's perception of the real environment is being augmented with virtual data. Hence HS is inherently Augmented Reality. An example of a real scenario where AR testing procedures are useful is in obstacle detection and avoidance in the underwater environment by an AUV. See *Obstacle detection and avoidance example in* Section 4.2.

Hardware-in-the-Loop (HIL) is another type of mixed reality testing technique. This type of testing allows the platform to be tested in a laboratory instead of in its intended environment. This is achieved by simulating all required exterioceptive sensors using a virtual environment. Virtual sensor data is then sent to the real platform's systems in order to fool them. In essence this is simply virtual reality for robots. Concurrently, the outputs of

higher level systems, which receive the simulated data, can be relayed back and displayed in the virtual environment for operator feedback. This can help show the system developer that the robot is interpreting the simulated sensor data correctly. HIL requires that all sensors and systems that interact directly with the virtual environment are simulated. Vehicle navigation systems are a good example since these use exterioceptive sensors, actuators and motors to determine position. Using simulated sensors means that the developer can specify exactly the data which will be fed into the systems being tested. This is complicated to do reliably in the real environment as there are too many external factors which cannot be easily controlled. Augmenting the virtual environment with feedback data from platform for observation means that HIL can be Augmented Virtuality as well as merely virtual reality for the platform.
Consequently, HIL and HS are both deemed to be Mixed Reality concepts, thus any testing architecture for creating the testing facilities should provide all types of mixed reality capabilities and be inherently distributed in nature.

3. ARF

The problem is not providing testing facilities as such, but rather being able to create them in a timely manner so that the costs do not outweigh the benefits. Any architecture for creating mixed reality testing scenarios should be easily configurable, extendable and unrestrictive so that it is feasible to create the testing facilities rather than do more expensive and less efficient real world tests. In essence, creating testing facilities requires a short term payout for a long term gain. Long term gains only applicable if the facilities are extendable and re-configurable for different tasks.
ARF is a component based architecture which provides a framework of components that are specifically designed to facilitate the rapid construction of mixed reality testing facilities. ARF provides a generic, extendable architecture based on JavaBeans and Java3D by Sun Microsystems. ARF makes use of visual programming and introspection techniques, to infer information about components, and consequently provides users with help via guided construction for creating testing scenarios. This allows for rapid prototyping of multiple testing combinations allowing virtual reality scenarios to be realised quickly for a multitude of different applications.
There are other architectures which provide HIL, HS and PS capabilities such as Neptune (Ridao et al., 2004). However, they only focus on testing and do not provide the extendibility and low level architecture that allows for easy extension using users own components. This is where ARF provides enhanced capabilities since it uses visual programming to provide a more intuitive interface allowing quick scenario creation and for configurations to be changed quickly and easily by abstracting the user from modifying the configuration files directly.

3.1 Visual programming

A subject that has been touched upon is that of visual programming or guided construction. One of the main performance inhibiting problems for software programmers and designers is whether or not they know exactly how all of the software modules they require work, and knowing how to use them. More often than not computer software modules are poorly documented and do not provide example implementations on how to use them. This is particularly the case when projects are on a tight schedule with little capital backing them,

since there simply isn't the time or the money to spend on creating nicely commented and documented code. This problem is self perpetuating since each time a badly documented module is required, the programmer spends so much time figuring out how to use the module that they then have less time to document their own code.

Poor documentation is merely one aspect which decreases productivity when it comes to developing software modules. Another problem for the programmer is knowing which modules are available and their functionality. Quite often package names and module names are not enough for the programmer to determine a module's functionality. Consequently, the programmer has to trawl through the API specification for the entire programming library to find out whether or not it is useful to them. Documentation maybe poor or non-existent, even if it does exist it can be time consuming to find out exactly what to do to use the module because no sensible examples are given. Thus, most of the time spent by the programmer is not spent actually programming. Conversely, when a programmer knows exactly the functionality of a module they can create a program to use it with great speed. Therefore, any architecture which reduces the amount of time spent by the programmer looking at documentation the faster they can finish the task.

This combination of problems means that programmers spend a lot of time re-inventing the wheel since existing modules are hard to locate, poorly documented or impossible to use. This problem is rife when it comes to producing virtual environments and simulated modules for testing robots, especially AUVs. This is usually due to environments being quickly “hacked up” to fulfil one purpose without considering the many other potential usages. Monolithic programming approaches then make reconfiguration and extension almost impossible. Add-ons can sometimes be “hacked” into the existing program, but in essence it is still a very inflexible program that will eventually become obsolete because a new usage or platform is required. At this stage it may be too complicated to try and extend the existing program to fulfil the new requirements, so instead a new program is quickly created with a few differences but is in essence just as inflexible as the first. More time spent making generic modules with basic inputs and basic outputs in a configurable environment makes making changes later on quicker and easier. However, when completely new functionality is required, a configurable environment still has to be reprogrammed to incorporate new modules. This can be difficult unless the environment is specifically designed to allow extension.

Visual programming (Hirakawa & Ichikawa, 1992) provides a solution for rapid module development and provides some ideas which can be harnessed to provide the basic idea behind a generic architecture for creating virtual environments. Visual programming is the activity of making programs through spatial manipulations of visual elements. It is not new and has been around since the very early 1970s when logic circuits were starting to be designed using Computer Aided Design (CAD) packages. Visual programming is more intuitive than standard computer programming because visual programming provides more direct communication between human and computer which given the correct visual queues makes it easier for the user to understand the relationships between entities, and thus makes connecting components easier. Consider the example of taking ice cube trays out of the freezer and placing one in a drink. The human way of doing this is simply to look to locate the ice cubes, use the hands to manipulate the ice cube out of the tray and then drop it into the drink. Thus any interface which allows the user to work in their natural way is going to make the job quicker. The other option, which is more like computer programming,

is to have the human write down every single action required to do this. The visual programming approach might be to manipulate a visual representation of the ice cube trays by dragging and clicking a mouse. Programming is far more clunky and will take much longer as it is not as intuitive. Therefore, visual programming aims to exploit natural human instincts in order to be as intuitive and effective as possible.

3.2 JavaBeans

Visual programming is a good idea, however, due to its visual nature it places requirements on how a module is written. This usually requires that the low level components be programmed in a specially designed language which provides more information to the visual programming interface. This leads to visual programming only being used for more specific uses, such as connecting data flows using CAD packages. However, visual programming can become far more powerful if it places nearly zero restrictions on how the low level components are created i.e. the programming language used. In order for visual programming to be widely accepted it has to some how make use of existing software components even if they are not designed to be used in this way. One such method of visual programming exists whereby components only have to implement a few simple "programming conventions" in order to be able to be used visually. These special software components are called JavaBeans and are based on the Java programming language (Sun Microsystems).

JavaBean visual programming tools work on the basis that a Java class object has been programmed adhering to certain coding conventions. Using this assumption the visual programming tool is able to use introspection techniques to infer what the inputs and outputs to a Java class and then display these as properties to the user. Thanks to Java's relatively high level byte code compilation layer, it is relatively simple for a JavaBean processor to analyse any given class and produce a set of properties which a user can edit visually. Therefore, removing the need for the programmer to write code in order to allow the configuration of Java class objects.

JavaBean programming environments currently exist which allow a user to connect and configure JavaBeans to make 2D GUI based applications. The BeanBuilder (https://bean-builder.dev.java.net) is one such program which provides the user with an intuitive visual interface for creating software out of JavaBeans. However, this doesn't provide any extra guidance other than graphical property sheet generation. A virtual environment is needed for mixed reality testing scenarios and this cannot be easily provided using the BeanBuilder's current platform. However, JavaBeans offer a very flexible base upon which a virtual environment development tool can be built. Since it can easily be extended via JavaBeans and all the advantages of JavaBeans can be harnessed. Another advantage of using JavaBeans is that scenario configurations can be exported to XML file for distribution to others and manual configuration.

3.3 Architecture

ARF provides the ability to execute all testing regimes across the reality continuum. It does this by incorporating: the OceanSHELL distributed communication protocol, vehicle dynamics & navigation simulators, sensor simulation, an interactive three-dimensional (3d) virtual world and information display. All spatially distributed components are easily interconnected using message passing via the communication protocol, or directly by method call using ARF's visual programming interface based on JavaBeans. The key to ARF's HIL and HS capabilities is the flexibility of the communications protocol. Other

external communications protocols are easily implemented by extending the event passing currently used by ARF's JavaBeans.
ARF provides a new type of JavaBean which allows the user to create a 3D environment out of JavaBeans. It is called a Java3DBean and is based on Java3D and JavaBeans. Java3DBeans inherently have all the functionality of JavaBean objects but with the added advantage that they are Java3D Scenegraph nodes. This gives them extra features and functionality such as 3D geometry, behaviours and able to interact with other Java3DBeans within the virtual world. ARF provides a user interface which extends the JavaBean PropertySheet allowing for Java3DBeans to be configured in the same way. The user is able to construct the 3D environment using Java3DBeans and decide which data to input/output to/from the real world. This provides unlimited functionality for HIL, HS and PS testing since any communication protocol can be implemented in JavaBeans and used to communicate to/from ARF to a remote platform. Mixed reality techniques can be used to render data visually in order to increase the situational awareness of an operator of a UUV, and provide simulation of systems for testing the remote platform. This increases the rate at which errors are detected, resulting in a more robust system in less time.

3.4 OceanSHELL distributed communications protocol

The obstacle detection and avoidance example *(see section 4.2)* highlights the need for a location transparent communication system. ARF requires real modules to be able to be swapped for similar simulated modules without the other systems knowing, having to be informed or programmed to allow it. The underlying communication protocol which provides the flexibility needed by the framework is OceanSHELL (Ocean Systems Lab, 2008). OceanSHELL provides distributed communications via UDP packets allowing modules to run anywhere i.e. provides module location transparency. Location transparency makes mixed reality testing straight forward because modules can run either on the remote platform, or somewhere else such as a laboratory.
OceanSHELL is a software library implementing a low overhead architecture for organising and communicating between distributed processes. OceanSHELL's low overhead in terms of execution speed, size and complexity make it eminently suited for embedded applications. An extension to OceanShell, called JavaShell, is portable because it runs on Java platforms. Both JavaShell and OceanShell fully interact, the only difference being that OceanShell uses C structures to specify message definitions instead of the XML files which JavaShell uses. However, both systems are fully compatible. OceanShell is not only platform independent but also language independent, increasing portability.
ARF allows for dynamic switching of OceanSHELL message queues and changing of port number. This allows for information flows to be re-routed by simulated modules in real time. This is ideal for doing HS or HIL testing. Figure 2 shows how OceanSHELL is used to communicate between the remote environment and the virtual environment.

3.5 ARF features

The Augmented Reality Framework (ARF) is a configurable and extendible virtual reality framework of tools for creating mixed reality environments. It provides sensor simulation, sensor data interpretation, visualisation and operator interaction with the remote platform. ARF can be extended to use many sensors and data interpreters specific to the needs of the user and target domain. ARF is also domain independent and can be tailored to the specific needs of the application. ARF provides modularity and extendibility by providing

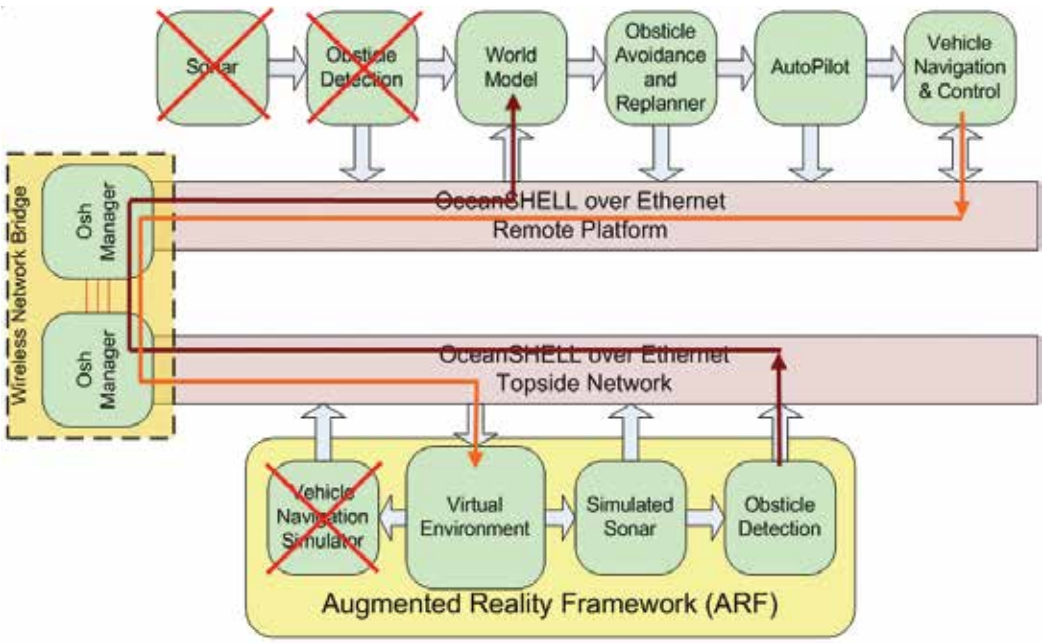


Fig. 2. This diagram shows how OceanSHELL provides the backbone for switching between real and simulated (topside) components for use with HS/HIL.

mechanisms to load specific modules created by the user, and provides a visual programming interface used to link together the different components. Figure 3 shows the ARF graphical user interface which the user uses to create their virtual environment. The 3D virtual environment is built using the scenegraph displayed in the top left of figure 3.

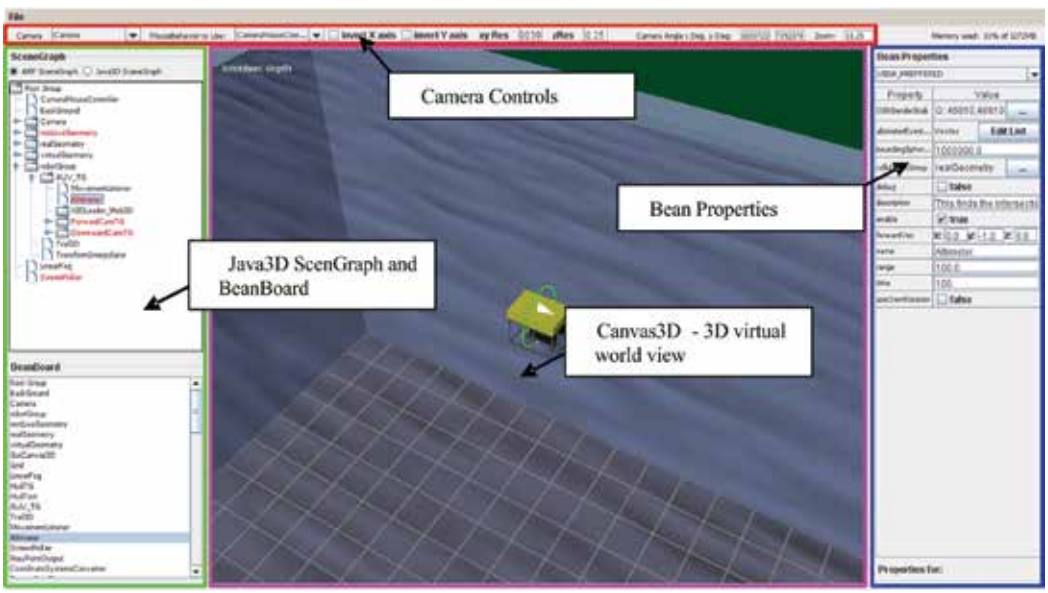


Fig. 3. ARF GUI Overview

ARF provides many programming libraries which allow a developer to create their own components. In addition ARF has many components ready for the user to create their own tailored virtual environment. The more ARF is used, the more the component library will grow, providing greater and greater flexibility, therefore exponentially reducing scenario creation times.

The ARF framework provides a 3D virtual world which Java3DBeans can use to display data and to sense the virtual environment. ARF provides many basic components to build virtual environments from. These components can then be configured specifically to work as desired by the user. If the required functionality does not exist the user can programme their own components and add them to the ARF component library. For example, a component could be a data listener which listens for certain data messages from some sensor, on some communication protocol (OceanSHELL, serial etc), and then displays the data "live" in the virtual environment. The component may literally be an interface to a communications protocol like OceanSHELL, from which other components can be connected in order to transmit and receive data.

ARF has the ability to create groups of configured components which perform some specific task or make up some functional unit. This group of components can then be exported as a super-component to the ARF component library for others to use. For example, an AUV super-component could include: a 3D model of an AUV, Vehicle Dynamics simulator, sonar, and a control input to the Vehicle Dynamics (keyboard or joystick). These virtual components can then be substituted for the real AUV systems for use with HIL and HS or copied and pasted to provide multiple vehicle support.

ARF allows complete scenarios to be loaded and saved so that no work is required to recreate an environment. ARF has components which provide interfaces to OceanSHELL, sensor simulation (sonar and video) and provides components for interpreting live OceanSHELL traffic and displaying it meaningfully in the virtual world. Figures 7 & 8 show a simple Sonar simulation using the ARF virtual environment. This sonar can then be output to a real vehicle's systems for some usage i.e. obstacle detection.

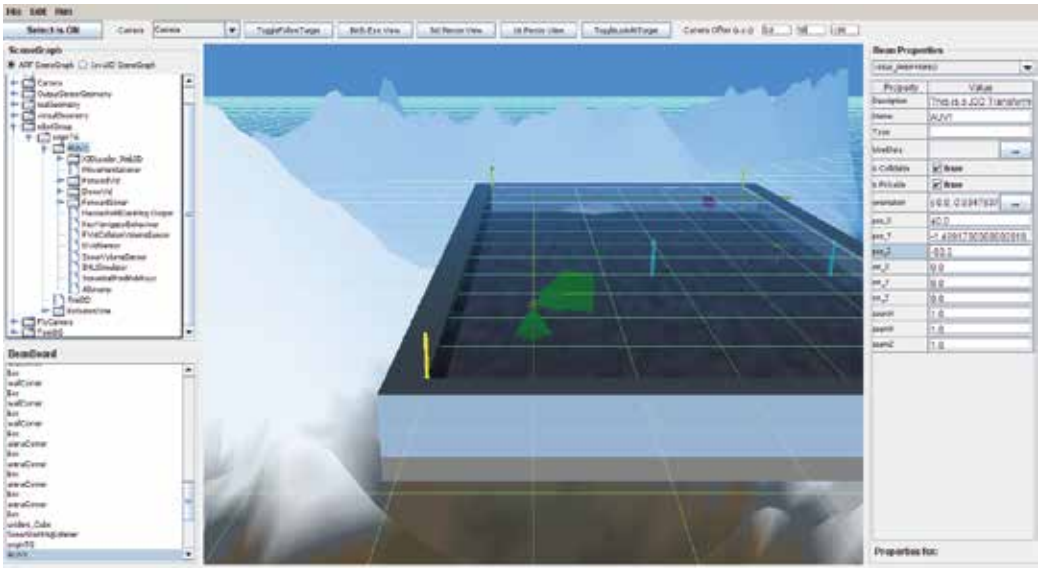


Fig. 4. Sauc-e AUV Competition Virtual environment for HIL testing 2007.

ARF provides a collection of special utility components for the user. These components are also provided in the programming API to be extended by the programmer to create their own JavaBeans. Java3DBeans are merely extensions to Java3D objects which adhere to the JavaBean programming conventions. ARF is capable of identifying which Beans are Java3DBeans and therefore knows how to deal with them. The only real difference between Java3DBeans and JavaBeans is that Java3DBeans are added to the 3D virtual world part of ARF as JavaBeans are only added as objects to the ARF BeanBoard (which keeps track of all objects). However, Java3DBeans can still communicate with any other objects in ARF's BeanBoard in the same way as JavaBeans.
In summary, JavaBeans are a collection of conventions which, if the programmer adheres to, allow a Java class to be dynamically loaded and configured using a graphical interface. The configurations of objects can also be loaded and saved at the click of a mouse button to a simple human readable XML file.
ARF provides many utility JavaBeans and Java3DBeans which the user can use directly, or extend. These include:

- Geometric Shapes for building scenes
- Mesh File loaders for importing VRML, X3D, DXF and many more 3D file types.
- Input listeners for controlling 3d objects with input devices (keyboard, mouse, joystick).
- Behaviours for making 3d objects do things such as animations.
- Camera control for inspecting and following the progress of objects.
- OceanSHELL input/output behaviours for rendering real data and for outputting virtual data from simulated sensors.
- Basic sensors for underwater technologies are provided such as forward looking sonar, sidescan sonar, bathymetric sonar, altimeter, inertial measurement unit (IMU), Doppler velocity log (DVL) etc.
- Vehicle Dynamics Models for movement simulation.

4. Applications of ARF

It is very hard to measure the effectiveness of ARF in improving the performance of creating testing scenarios. Performance Testing (see section 5) alone does not reflect how ARF is likely to be used and also does not demonstrate ARF's flexibility either. Although the potential applications are innumerable, this section describes some representative examples of applications and topics of research that are already gaining benefits from the capabilities provided by the Augmented Reality Framework.

4.1 Multiple vehicle applications

The main objective of the European Project GREX (http://www.grex-project.eu) is to create a conceptual framework and middleware systems to coordinate a swarm of diverse, heterogeneous physical objects (underwater vehicles) working in cooperation to achieve a well defined practical goal (e.g. search of hydrothermal vents) in an optimised manner.
In the context of GREX, algorithms for coordinated control are being developed. As these algorithms need to be tested on different vehicle platforms (and for different scenarios), real testing becomes difficult due to the cost of transporting and using vehicles; furthermore, the efficiency and safety of the different control strategies needs to be tested. The ARF virtual

environment provides the ideal test bed: simulations can be run externally and fed into the virtual AUVs, so that the suitability of the different control strategies can be observed. The virtual environment serves not only as an observation platform, but can be used to simulate sensors for finding mines as used in the DELPHÍS multi-agent architecture (Sotzing et al., 2007), depicted in Figure 6.

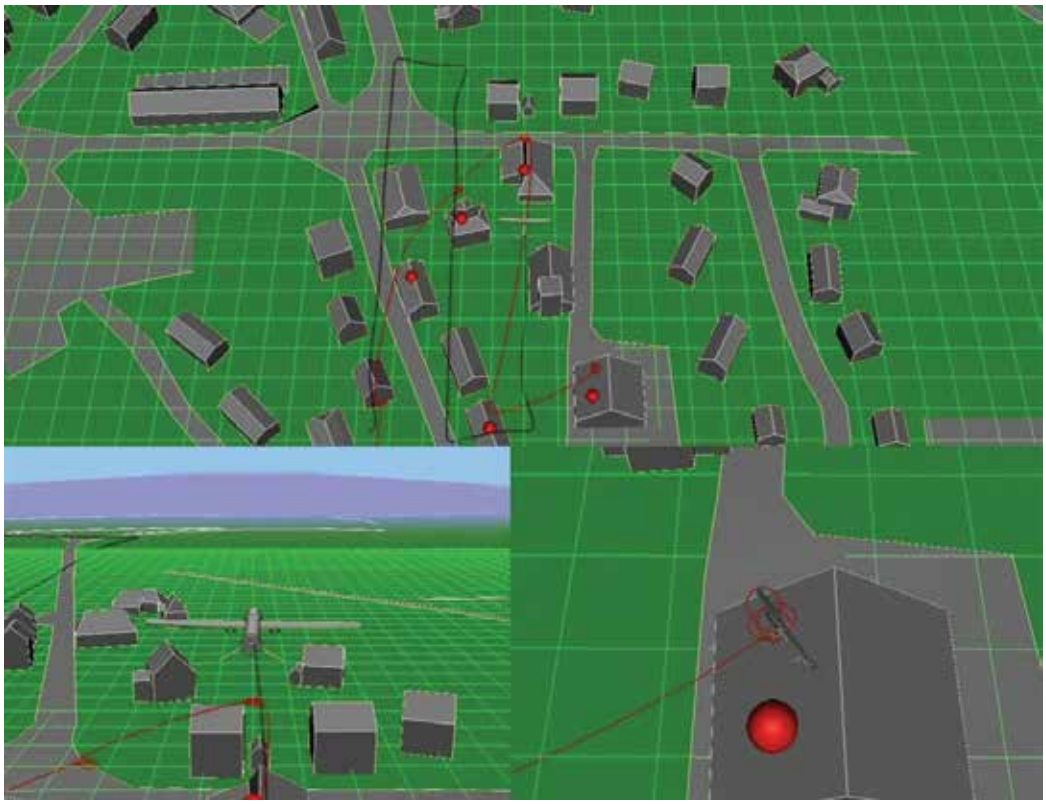


Fig. 5. Simulated UAVs cooperating and collaborating to complete a mission more efficiently.

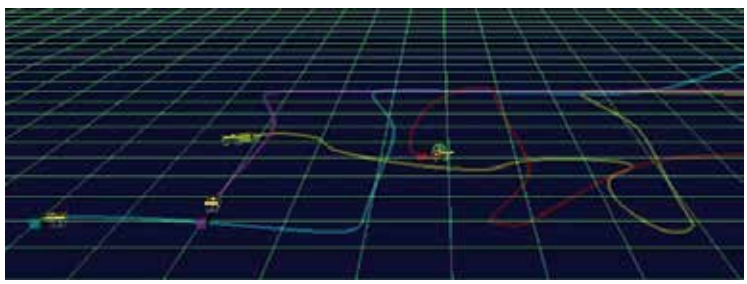


Fig. 6. Multiple AUVs running completely synthetically cooperating and collaborating to complete a mission more efficiently.

Other applications of the DELPHÍS multi agent architecture have been demonstrated using ARF. These include a potential scenario for the MOD Grand Challenge. This involves both surface and air vehicles working together to find targets, in a village, and inspect them. DELPHÍS was tested using simulated air vehicles, rather than underwater vehicles, which executed a search, classify & inspection task. ARF provided the virtual environment, vehicle simulation and object detection sensors required for the identification of potential threats in the scenario. Figure 5 displays the virtual environment view of the Grand Challenge scenario. The top of the screen shows a bird's eye observation of the area with the bottom left & right views following the survey class unmanned aerial vehicle (UAV) and inspection class UAV respectively. The red circles represent targets which the survey class UAV will detect upon coming within range of the UAV's object sensor. Information regarding specific targets of interest is shared between agents utilising the DELPHÍS system. Vehicles with the required capabilities can opt to further investigate the detected objects and reserve that task from being executed by another agent. Figure 6 shows AUVs working together to complete a lawn mower survey task of the sea bottom. The DELPHÍS system executes this quicker than a single AUV since each AUV agent does a different lawn mower leg. The agents appear to use a divide and conquer method, however, this is achieved completely autonomously and is not pre-programmed. The AUVs decide which lawn mower legs to execute based on distance to leg, predictions as to what other agents will choose, and which tasks have already been reserved by other competing self interested agents.

Apart from ARF being used for observation purposes, multiple AUVs/UAVs were simulated which helped tremendously for observing behaviours and testing of the DELPHÍS architecture. Basic object detection sensors provided a simple but effective method of outputting data from the virtual environment to DELPHÍS. Detections of certain types of objects meant that new goals were added to the plan. Generally these would be of the form: identify an object with one vehicle, then classify that object with another type of vehicle with the appropriate sensor. Thus some of the simulated AUVs were only capable of detecting the objects, whilst others were capable of inspecting and classifying those objects.

4.2 Obstacle detection and avoidance

One of the most common problems for unmanned vehicles is trajectory planning. This is the need to navigate in unknown environments, trying to reach a goal or target, while avoiding obstacles. These environments are expected to be in permanent change. As a consequence, sensors are installed on the vehicle to continuously provide local information about these changes. When object detections or modifications are sensed, the platform is expected to be able to react in real time and continuously adapt it's trajectory to the current mission targeted waypoint.

Testing these kinds of adaptive algorithms requires driving the vehicle against man-made structures in order to analyse its response behaviours. This incurs high collision risks on the platform and clearly compromises the vehicle's survivability.

A novel approach to this problem uses ARF to remove the collision risk during the development process. Using Hybrid Simulation, the approach uses a set of simulated

sensors for rendering synthetic acoustic images from virtually placed obstacles. The algorithms are then debugged on a real platform performing avoidance manoeuvres over the virtual obstacles in a real environment. Figure 2 shows the required framework components, Figure 7 shows the virtual environment view and Figure 8 shows the resulting simulated sonar of the obstacles. It should be noted that the topside simulated components can be switched on to replace the remote platform's real components, therefore achieving HIL or HS. A detailed description of the evaluation and testing of obstacle avoidance algorithms for AUVs can be found in Pêtrès et al. (2007) & Patrón et al. (2005).

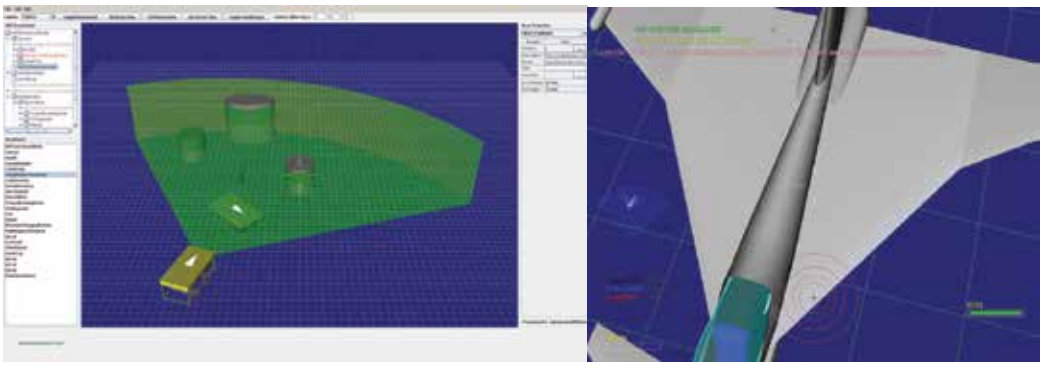


Fig. 7. ARF simulating Forward-look sonar of virtual objects.

Fig. 8. The resulting images of the simulated Forward look sonar.

4.3 Autonomous tracking for pipeline inspection

Oil companies are raising their interest in AUV technologies for improving large field oil availability and, therefore, production. It is known that Inspection, Repair and Maintenance (IRM) comprise up to 90% of the related field activity. This inspection is clearly dictated by the vessels availability. One analysis of potential cost savings is using an inspection AUV. The predicted savings of this over traditional methods for inspecting a pipeline network system are up to 30%.

Planning and control vehicle payloads, such as the AUTOTRACKER payload (Patrón et al., 2006), can provide such capabilities. However, as mentioned, vessel availability and off-

shore operation costs make these types of payloads a difficult technology to evaluate. ARF can provide simulated sidescan sonar sensors for synthetic generated pipeline rendering. These capabilities provide a transparent interface for the correct and low cost debug of the tracking technologies. Furthermore, potentially complicated scenarios, such as multiple pipeline tracking and junctions of pipes, can be easily created to test the pipe detection and decision algorithms of the AUTOTRACKER system (see figure 9). This could not easily be tested in the real environment as real-time debugging is not available and the potential for incorrect decision due to confusion about which pipeline to follow is high, therefore higher risk of loss of an AUV.

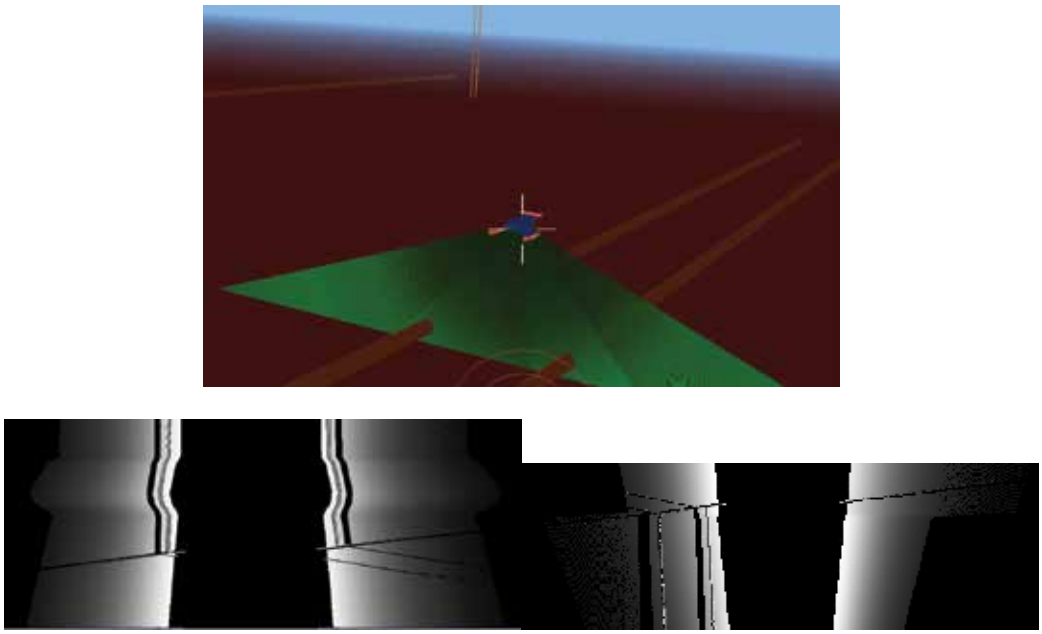


Fig. 9. Left & Right Sidescan Sonar simulation using ARF.

4.4 Nessie III

Nessie III (see figure 10) is the Ocean Systems Lab's entry to the 2008 Student Autonomous Underwater Vehicle Competition Europe (SAUC-E). The tasks the AUV had to execute included:

- Searching for the ground targets (tyres, drop-target, cones)
- Searching for the mid-water targets (orange and green balls).
- Touching orange ball
- Dropping weights on bottom target
- Surfacing in surface zone designated on the tangent between two tyres or two cones.

Specific behaviour routines are needed for lining up with the drop target for dropping the drop weights and for lining up with the orange ball ready for touching. This is straightforward if object positions are accurate and vehicle navigation doesn't drift. However, object positions are subject to inaccuracies due to sensor accuracy, false detections

and navigational drift. Thus once an object has been detected, take for example the orange ball, the mission planner will change its task to touching the orange ball so that the positional drift is minimal. Rather than searching for all objects first and then going back for a closer inspection later. Positions of detected objects are stored by a world model on Nessie, so that once higher priority tasks are completed, the AUV goes to roughly where it thinks that object should be and starts a new search pattern to relocate it. Figure 10 shows the AUV homing in on the orange ball to touch it. ARF is used to visualise what the robot thinks is happening. ARF plots the detection of the orange ball and moves the virtual AUV accordingly.

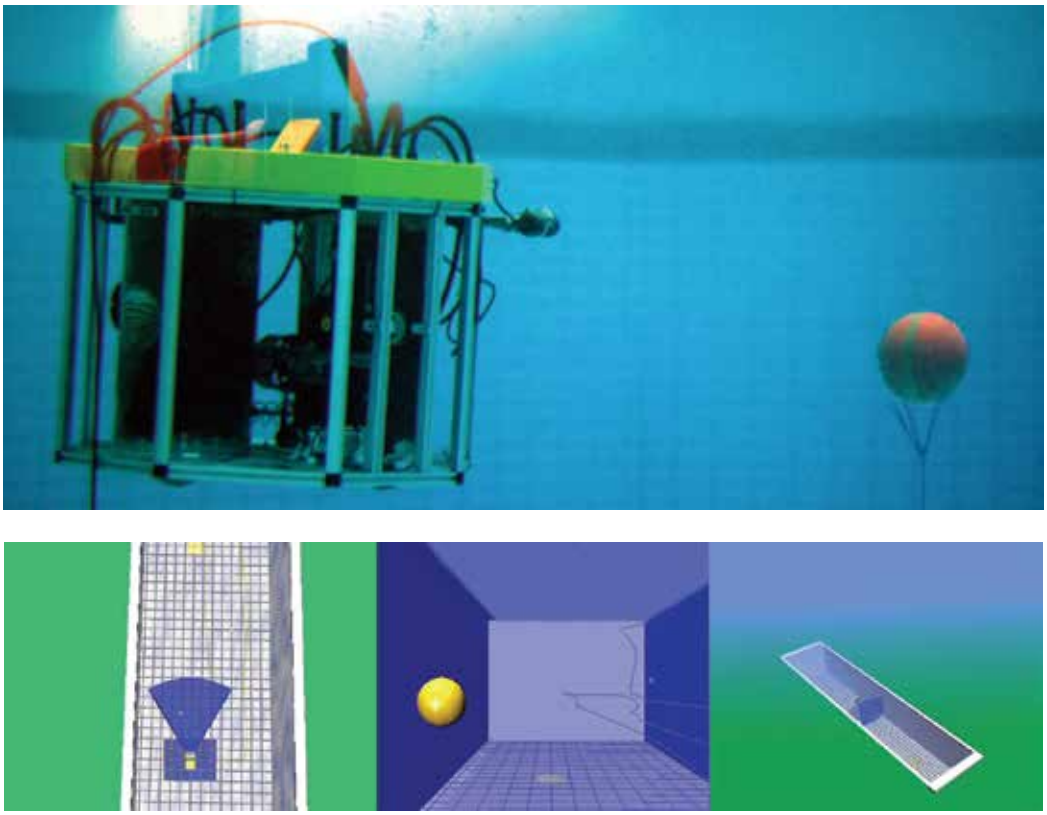


Fig. 10. Nessie tracking the orange ball for real and in simulation.

The Nessie III platform relies on visual tracking of the ground in order to estimate its movement and absolute position. In addition Nessie III is equipped with a Doppler Velocity Logger (DVL) which, combined with a compass, provides accurate navigation when visual systems cannot be used. Visual systems rely on the water being clear and well illuminated, but this is rarely the case in real environments. In addition to visualisation purposes ARF was used to test the behaviour of the autopilot, mission planner and Simultaneous Localisation and Mapping (SLAM) systems. A hydrodynamic model was used to simulate

the movement of Nessie and a simple object detection sensor was programmed with the same parameters as the real vision systems of Nessie III. This enabled the real platform to execute its mission using the real autopilot and, simultaneously, test the mission planner to see if it made the right decisions and did the correct search behaviours under different circumstances. The output of the detection sensors was used to help position the AUV when the same object was re-detected using SLAM. However, if the vision system which provided the ground tracking on Nessie III failed the only source of navigation would be dead reckoning. This alone was not accurate enough to collide with the targets in the competition. Therefore, for the case of the orange ball especially, visual servoing could be used home in on the ball and bring the AUV into contact with it. Tank testing time was limited so ARF was used to tune the visual servoing method of the autopilot. The bottom left 2 images in figure 10 show the field of view of the forward looking cameras and the orange ball which is being tracked.

5. Performance testing

Use-cases demonstrate ARF's many different uses, and how extendable it is due to the JavaBean component based architecture. The performance increase of using ARF to create testing scenarios whilst taking into account its performance optimisations, such as guided construction, rapid configuration and large code reuse, is unknown. This can be quantified by comparing it to the standard approach of programming a test environment by hand. Since ARF uses JavaBean classes, it is relatively straightforward to compare the speed at which scenarios can be created using ARF against scenarios created using the same JavaBean classes but programmed by hand.

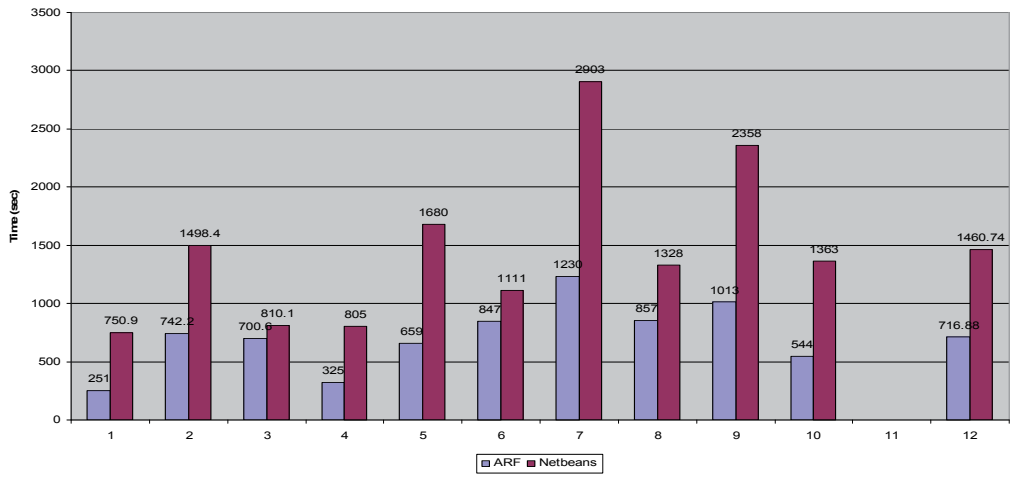


Fig. 11. Scenario creation times: ARF vs. Netbeans

For a fair test, the exact same JavaBean classes which are used in ARF will be used by the programmers. The programmers will use the NetBeans IDE as this provides help when

programming by showing the user relevant documentation of methods for classes. Also a full API specification will be available to programmers. The main task is to create a small virtual environment and connect the components together to simulate a basic AUV. The exact same components should be connected via programming using NetBeans. Measures of performance are gained from statistical feedback from the test participants as well as the tasks being timed. To make the test more useful, all test participants will be novices in using ARF and have only seen a brief video demonstration of how to use it, but are experts in programming. Other performance indicators, such as the amount of help and documentation required, are also logged. Figure 11 clearly shows the increase in performance of ARF compared with Netbeans. Number 12 on the graph shows the average values for all test participants which equates to on average ARF being over twice as fast as Netbeans. Generally, ARF would be expected to be even faster, however, because users were unfamiliar with ARF they took longer to do certain tasks. The times given above were to create a simple scenario with an AUV. Due to ARF's SuperComponent creation tools, the users were able to save their AUV configuration and then add a clone to the same environment and then adjust the properties so it was slightly different. The average time taken by the users to do this in ARF was *47 seconds*. Furthermore, the average time taken to make an alteration and save as a new configuration using ARF was only *37 seconds* compared with *77 seconds* in NetBeans. The ability to reuse large sections of code easily via the use of SuperComponents supports the "create once, use many times" ideal.

All these small increases in performance lead to greater increases in performance over a longer period of sustained use of ARF. The availability of such testing facilities helps the platform being developed become more mature within a shorter time frame. Different vehicle configurations can be easily created allowing for testing of potentially new vehicle configurations to see if they are viable and thus helping to develop the optimal solution of vehicle for a specific task.

7. Conclusions and further work

The testing carried out by the ARF Performance Tests (APT) highlight the features of ARF which are difficult to prove with only use cases. The most important points to draw from the results discussion is that ARF provides an extendable architecture which is generic enough to provide the capabilities of all the different usages. In essence ARF is a one size fits all architecture which provides generality on different granular levels providing flexibility for both programmers and high end users. The second most important feature that the APTs highlighted is the considerable performance improvement over conventional methods of scenario creation. Furthermore, the trade off of using ARF is minimal due to being built on existing JavaBeans technology so the programmer has little work to do to interface with ARF. The re-configurability of ARF is fast due to the use of simple project creation, SuperComponents and Guided Construction, which all help to increase programmer efficiency.

The analysis of the performance of ARF provides the fundamental evidence for why ARF is so powerful for higher level applications, such as HIL testing scenarios. The many small time savings which ARF provides propagate to massive time savings in more complex

projects. Scenarios no longer need to be programmed from scratch, they can simply be extended, have new components bolted on and merged with functionality from completely different projects through the use of SuperComponents. As ARFs Java3DBean and JavaBean component base grows, the more quickly scenarios can be created with ever more useful SuperComponents. The run away effect of having a large ever increasing component base means that before long design of new components will hardly ever be necessary. Further work is being carried out in increasing ARF's component base and expanding the usages of ARF into domains and fields other than sub-sea.

8. References

Choi, S.K.; Yuh, J. (2001).A virtual collaborative world simulator for underwater robots using multidimensional, synthetic environment. *In Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on Volume 1, 2001.* pp.926-931

Healey, A. J.; Pascoal, A. M.; Pereira, F. L. (1995) Autonomous underwater vehicles: An application of intelligent control technology: *Proceedings of the American Control Conference, Seattle, Washinton June 21-23, 1995,* pp. 2943-2949

Hirakawa, M.; Ichikawa, T. (1992). Advances in visual programming, *Proceedings of the Second International Conference on 15-18 June 1992 ICSI '92,* pp. 538-543

Lane, D.M.; Falconer, G.J.; Randall, G.; Edwards, I. (2001) Interoperability and Synchronisation of Distributed Hardware-in-the-Loop Simulation for Underwater Robot Development: Issues and Experiments. *Proceedings of the 2001 IEEE International Conference on Robotics and Automation, Seoul, Korea, May 21st – 26th*

Ocean Systems Lab. (2008). *OceanSHELL: An embedded library for Distributed Applications and Communications.* Heriot-Watt University

Patrón, P.; Smith, B.; Pailhas, Y.; Capus, C. & Evans, J. (2005). Strategies and Sensors Technologies for UUV Collision, Obstacle Avoidance and Escape: *7th Unmanned Underwater Vehicle Showcase September 2005*

Patrón, P.; Evans, J.; Brydon, J. & Jamieson, J. (2006). AUTOTRACKER: Autonomous pipeline inspection: Sea Trials 2005: *World Maritime Technology Conference - Advances in Technology for Underwater Vehicles, March 2006*

Pêtrès, C.; Pailhas, Y.; Patrón, P.; Petillot, Y.; Evans, J. &Lane, D.M.; (2007) Path Planning for Autonomous Underwater Vehicles: *proceedings of IEEE Transactions on Robotics, April 2007*

Ridao, P.; Batlle, E.; Ribas, D.; Carreras, M. (2004). NEPTUNE: A HIL Simulator for Multiple UUVs: *In proceedings of OCEANS '04. MTS/IEEE TECHNO-OCEAN '04 Volume 1, 9-12 Nov. 2004 pp.524-531*

Sotzing, C.C.; Evans, J. & Lane, D.M. (2007). A Multi-Agent Architecture to Increase Coordination Efficiency in Multi-AUV Operations: *In proceedings IEEE Oceans 2007, Aberdeen;*

Sun Microsystems, JavaBeans, *http://java.sun.com/javase/technologies/desktop/javabeans*

Zyda, M.J.; McGhee, R.B.; Kwak, S.; Nordman, D.B.; Rogers, R.C.; Marco, D. (1990). Three-dimensional visualization of mission planning and control for the NPS

autonomous underwater vehicle: *Oceanic Engineering, IEEE Journal Volume 15, Issue 3,* pp.217–221

6

Bio-inspired Robotic Fish with Multiple Fins

Parasar Kodati and Xinyan Deng
University of Delaware
U.S.A.

1. Introduction

In order to improve the performance of AUVs in terms of efficiency and maneuverability, researchers have proposed biomimetic propulsion systems that swim using flapping fins rather than rotary propellers. This calls for the exploration of unique locomotion characteristics found in a variety of fish for use in underwater robots. (Sfakiotakis et. al., 1999) present a good review of fish swimming modes targeted at roboticists interested in aquatic locomotion. A classification scheme of fish locomotion mechanisms, originally presented in (Lindsey, 2006) and was modified in (Colgate & Lynch, 2004), is shown in Fig. 1. The three main swimming styles are characterized by undulatory body motion, undulatory fin motion, and oscillatory fin motion. A more traditional classification is one proposed by Breder (Breder, 1926) that broadly identifies two styles of swimming: one is Body and/or Caudal Fin (BCF) locomotion, and the other is Median and/or Paired Fin (BMP) locomotion. Fish classes that use varying degrees of body undulation and/or caudal fin oscillations for thrust generation are examples of BCF swimming, and fish that use paired fins like the left and right pectoral fins, dorsal, and ventral pelvic fins for thrust generation are classified under the MPF swimming style.

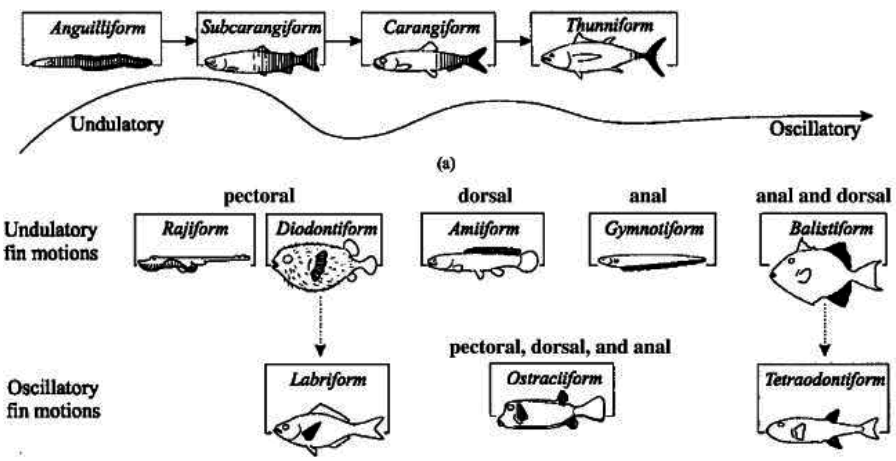


Fig. 1. Fish Classification based on swimming styles (Lindsey, 1978 & Colgate, 2004).

MIT's Robotuna marked the beginning of the biomimetic approach in underwater vehicles (Triantafyllou & Triantafullou, 1995). Barrett et. al. (Barrett, et. al, 1999) demonstrated that

the highly articulated robotic fish experienced less drag with undulatory motion than that seen without body undulation. Zhu (Zhu, et. al, 2002) also identified a vorticity control phenomenon, which explains the interaction of the vortices shed by the undulating body and the ones in the wake shed by the tail fin. Such interaction of the body with the wake was also shown to reduce the muscle activity in fish (Liao, et. al, 2003). Anguilliform requires a greater amount of body undulation and thus more degrees of freedom must be connected in series to form a robot. McIsaac and Ostrowski (McIsaac & ostrowski, 2003) studied motion planning and control of a serial chain robotic eel. They generated gaits (time functions of the joint angles) for the straight and turning motions of 3-link and 5-link robots. MacIver (MacIver, et. al, 2004) presented some aspects of underwater vehicle design in the areas of sensing and motion mechanisms of a knifefish in (Epstein, et. al, 2006).

The Biologically Inspired Robotics Group at EPFL presented a swimming and crawling robot, BoxyBot (Lachat, et. al, 2006) that is "loosely inspired by the boxfish". The focus there was to mimic boxfish-like switching of swimming modes under different speed ranges. More recent biologically inspired robot designs include Basilisk lizard like water running robot (Floyd, et. al, 2006).

Recently, a new class of biorobotic underwater vehicles based on the biomimetic principles of flapping foils are being designed. These platforms employ fin designs and motion kinematics that are the result of experimental and computational fluid mechanics work. Licht et. al. (Licht, et. al, 2004) presented the design of a vehicle platform with four heaving and pitching foils. A team of ocean engineers, fluid mechanists and biologists proposed a concept vehicle taking advantage of high efficiency foils in combination with articulated pectoral fins with rays for enhanced maneuverability (Fish, et. al, 2003). Pulsatile jet formation loosely inspired by squid is presented in (Mohseni, 2004). Kato's Bass III (Kato & Wicaksono, 2000) is the latest 3 DOF pectoral fin based vehicle designed for low speed precise maneuvering. Bandyopadhyay (Bandyopadhyay, 2005) presented a comprehensive review of approaches on various fronts of biomimetic underwater vehicle technology like high lift generating fin hydrodynamics, vehicle maneuverability using pectoral fins, muscle-like actuators and neuroscience based control.

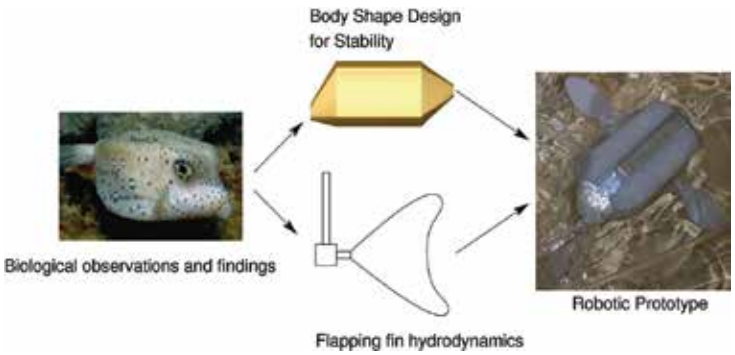


Fig. 2. Bio-inspired robotic fish with multiple fins.

Most Autonomous Underwater Vehicles (AUVs) today are larger ones applied extensively from environmental monitoring to oil and gas exploration (Yuh, 2000). However, these AUVs are not suitable for applications where the vehicle has to explore confined spaces like ship wrecks or oil pipe lines, where maneuverability and stability are more important than speed. Tasks such as these call for designs that are small, maneuverable and precisely

controlled. The work presented in this chapter is a step towards realizing such Micro Underwater Vehicles(MUVs). Here we present the research leading to the design and fabrication of a bio-inspired robotic fish with multiple fins for propulsion and control [Fig.2]. We present the experimental setup for studying flapping fin hydrodynamics and the force measurement results from flexible fins, and we presents the modeling of boxfish-like engineering shapes incorporating key morphological features that are responsible for self correcting vorticity generation as explained by Bartol et. al. in (Bartol, et. al, 2005).

2. Fin hydrodynamics

A robotic flapper was designed to generate fin motion in three independent rotational degrees of freedom, with a force/torque sensor attached to the base of the fin. The flapper was mounted on a linear stage that was driven a stepper motor. A larger tank was used to allow for linear motion. The tank was filled with refined paraffinic oil. This setup allows us to reply the fin kinematics while measuring instantaneous hydrodynamic forces.
The thrust (force in the direction of motion) produced by a flapping fin is a result of its interaction with the surrounding fluid. In the case of simple flapping (for example the caudal fin of a boxfish in cruise mode) this interaction can lead to lift and drag based forces as well as added mass effects (resulting from the deceleration of fluid set in motion by previous strokes). Fig. 3 shows such forces acting on a foil. The lift component (L) of the total force (F) is a result of periodic vortex shedding while the rest of the components are due to drag(D).

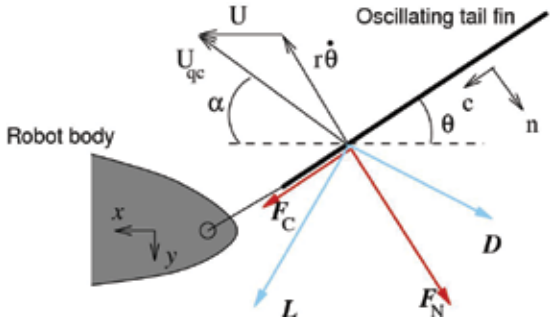


Fig. 3. Forces acting on an oscillating tail fin.

In this study, a boxfish like fin shape was used as the template to vary the fin geometry, therefore its flexibility. To change the chord wise flexibility, the dimensions c1 and c2 have been varied while fixing the aspect ratio and the total area of the fin. This kind of parametric variation displaces the center of pressure of the fin, thus varying the degree of flexing. Fig. 4 shows the shape template used to obtain different tail fin shapes by varying c1 and c2. Four shapes (labeled S1 to S4) from this continuum have been considered experimentally. Table 1 gives the geometric dimensions. The shapes were cut from 0.6mm thick Delrin, 0.1mm thick Polyimide, and 0.6mm thick Polyethylene in order to look at them over a range of material stiffness. In fact, from a flexibility point of view all twelve shapes (four profiles, three materials) form a continuum of flexural stiffness (EI) values, which have a strong correlation to the amount of flexing (Combes & Daniel, 2003). The term captures the elasticity (E, Young's modulus) of the fin material as well as the shape geometry (I, second moment of area) that together determine the fin's flexibility.

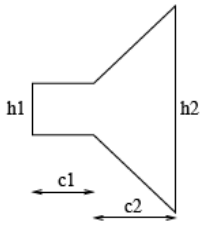


Fig. 4. Fin shape parameters.

Shape	C1 (cm)	C2 (cm)
S1	4.00	4.11
S2	3.01	4.57
S3	2.02	5.02
S4	1.00	5.50

Table 1. Four shapes studied.

Each shape/material combination (with the exception of shape S1 of Polyethylene) was flapped harmonically at frequencies ranging (in 0.1 Hz intervals) from 0.3Hz to 0.8 Hz, while being towed at a speed of 0.08 m/s. The Strouhal number is a non-dimensional number that relates the forward velocity, U to the flapping frequency,f as:

$$St=fw/U \tag{1}$$

where w is the wake width. In our case, w was treated as the width of a single fin stroke. We define the non-dimensional thrust and drag coefficients as

$$C_T = 2F_x/(\rho A U^2) \tag{2}$$

$$C_D = 2F_y/(\rho A U^2) \tag{3}$$

where Fx and Fy are forward thrust and side way components of the force vector in the horizontal plane, ρ is the density of the fluid and A is the fin area. η is defined as a measure of forward propulsive efficiency

$$\eta = kF_xU/(M_z\omega) \tag{4}$$

The denominator in Equation 4 is taken as a measure of the torsional power required to drive the fin where Mz is the moment measured by the force sensor and ω (= 2πf) is the circular frequency of flapping. k is a non-dimensional scaling constant.

Fig. 5 shows the variation of C_T over the four fin shapes of different materials (Delrin, Polyimide, and Polyethylene). C_T increased with frequency except for Shape S1 of Polyimide and Shape S2 of Polyethylene fins. These two fins have a drastic drop in C_T after a certain frequency and then it begins to increase again. One possible explanation for this is that the force generation at lower frequencies for these two fins is viscous drag dominant. This effect drops after the Reynolds number, contribution of added mass, and pressure drag start to increase. Fig. 6 shows the variation of η (measure of efficiency) with Strouhal number for all the fins used. The relatively more rigid Delrin fins have peak efficiencies in the range of 0.8-1. Polyimide fins do not have much variation, for most part, with Strouhal number. The low efficiency of the Polyethylene fins can once again be explained by high viscous drag.

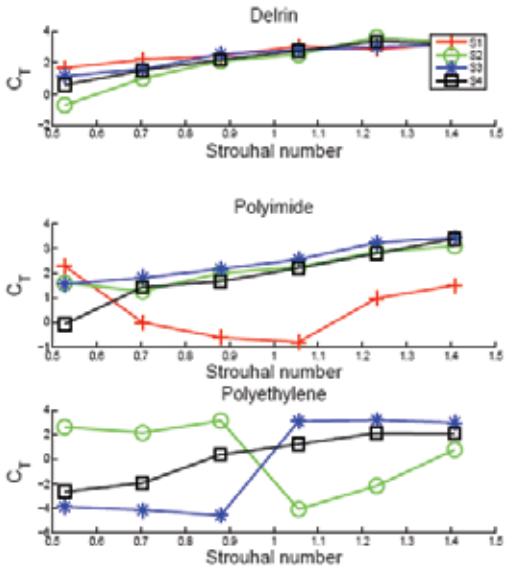


Fig. 5. Variation of thrust coefficient with frequency.

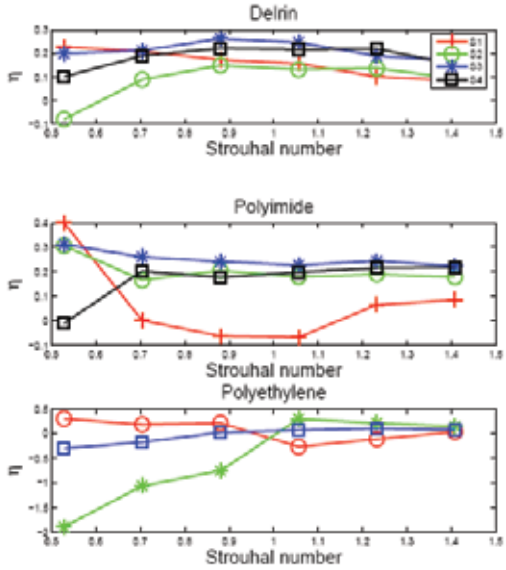


Fig. 6. Variation of efficiency with frequency.

The criteria for fin selection should be high values for C_T and η over a range of frequencies. Caudal fin flapping frequency is one of the key parameters that has to be changed to control the speed of the MUV. Thus, a fin with good thrust production and high efficiency over a range of frequencies is desirable. Shape S3 of Polyimide clearly is the best choice as indicated by the C_T and η values.

3. Body stability

This study seeks to understand the role of different morphological parameters that could be responsible for boxfish stability and transform such findings into engineering design guidelines for the body shape design of a micro underwater vehicle. The analysis was performed using Computer Aided Design and Engineering (CAD/CAE) tools such as solid modeling software and fluid flow simulation software. First approximate models of boxfish were built in 3D modeling software and fluid flow was simulated over such models to analyze the vorticity patterns around the body. The relative role of key morphological features like the dorsal and ventral keels, concavity and convexity of the shape, and changes in the cross section along the length of the body were determined. Different MUV body shape designs were built along these lines and tested for required vortex strength and overall drag to arrive at the best design.

Fig. 7 shows the counter rotating vortex shedding by a bluff body. The vortex shedding should result in moments that can correct disturbances in the pitch and yaw(not shown in the Figure) directions. The shape that can demonstrate this 'selfcorrecting vortex shedding' for pitch and yaw disturbances will be the shape suitable for the body of the MUV.

The morphological features that contribute to the unique vortex shedding patterns as reported in (Bartol, et. al, 2005) had to be identified to be incorporated into the design. Approximate 3D models capturing essential features of the boxfish were made and flow at different angles of attack was simulated over such models to study the role of the various features on stability. SolidWorksR , a 3D CAD modeling software was used to build body shapes and fluid flow simulation was carried out using GAMBIT ™ (for defining boundaries of the flow and laying computational grid around the body) and FIDAP™ (for solving the flow and post processing) of FLUENT Inc.

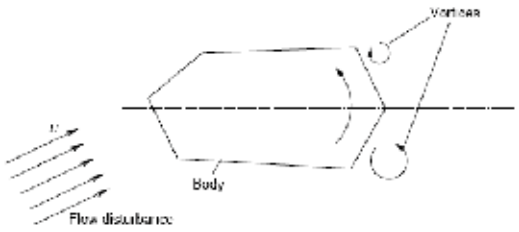
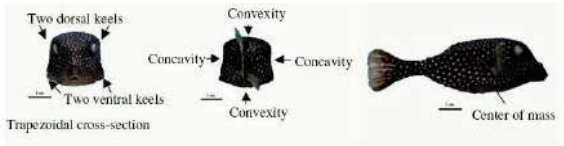


Fig. 7. Self-correcting vortex shedding in boxfish body.

Fig. 7. Cross sections along the body of the buffalo trunk fish.

3D models of boxfish were built in SolidWorks. Essential features like dorsal and ventral keels and variation in cross sections along the length of the body were reproduced in the models. Variation of cross section along the body plays a very significant role in the vorticity

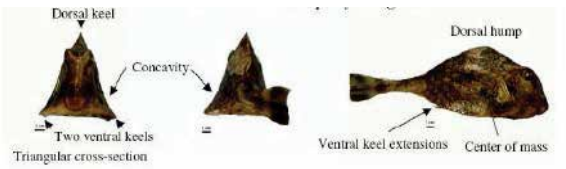
generation: therefore, cross sections near mouth, eye ridge, peduncle, and any other distinct plane were drawn on 2D planes and joined to obtain the desired shape. Fig.7 shows the planes containing important cross sections of the buffalo trunk fish. Only two, the Spotted boxfish and the Buffalo trunkfish, of the four varieties of boxfish studied in (Bartol, et. al, 2005) were considered here. The other two shapes are not significantly different from the buffalo trunkfish. Fig. 8(a) and 9(a) show the planar views of the actual boxfish used in the study by Bartol et. al. 3D models of the fish used in this study are shown in Fig. 8(b) and 9(b). It has to be noted that the 3D models were developed only based on subjective 'resemblance', capturing the key features and not accurate measurements of the boxfish morphology.



(a) Planar views of the Spotted boxfish considered by Bartol et al. in [11]

(b) CAD Models

Fig. 8. Spotted boxfish



(a) Planar views of the Spotted boxfish considered by Bartol et al. in [11]

(b) CAD Models

Fig. 9. Buffalo trunkfish

Once created, the 3D models were imported into GAMBIT in IGES format. The imported models were then 'cleaned' by merging unwanted edges and surfaces that may have hindered mesh generation. To take advantage of the lateral symmetry of the boxfish shape, only (left/right) half of the body was considered for simulation. An external brick volume was used for the fluid flow. The symmetry plane of the boxfish was made coincident to one of the brick side walls and the body was placed midway along the height and length of the brick (see Fig. 15). In future simulations, the size of the brick could be increased further to minimize the wall effects. Fig. 15 shows the 3D mesh generated in GAMBIT. Shape functions have been defined such that the mesh is finer near the body surfaces and gets coarser away from the body. The mesh size was kept fairly coarse in order to keep the

convergence time reasonable. The course mesh was not detrimental to the simulation results because the Reynolds number was only about 300, and thus did not require a very dense mesh. It has to be noted that the Reynolds number corresponds to the case where a water wave disturbs an otherwise stationary boxfish and therefore is lower than the average Reynolds number of boxfish swimming.

Here the main results and conclusions are summarized and compared to those reported in (Bartol, et. al, 2005). First, it was reported in (Bartol, et. al, 2005)that the ventral keels of all the models produced Leading Edge Vortices (LEVs) that grew in circulation along the bodies, and this was verified in the present study. Vorticity concentration was found at the keel edges for all the models at various cross sections. Vorticity contours at different cross sections for both the fish can be found in (Kodati, 2006). This is expected, as the keels form the sharp corners of the body and hence induce circulation into the oncoming water. However, for a given angle of attack an increase in circulation was found only when there is a sudden increase in the cross section. This is true for both the boxfish shapes. For the spotted boxfish in Fig. 10, the maximum concentrated vorticity is located at the eye ridge, where the cross section increases suddenly. Similar behavior was observed for the Buffalo trunkfish. The concentrated vorticity near the keels increased along the length until it reached a maximum at the mid body plane, where the body curve attains a peak (Fig. 11).

Second, the present study also verified that vortices formed "above the keels and increased in circulation as pitch angle became more positive, and formed below the keels and increased in circulation as pitch angle became more negative", verbatim from (Bartol, et. al, 2005). Fig. 11 shows the net vertical lift on the spotted boxfish model varying with pitch angle. In (Bartol, et. al, 2005) the lift coefficients of all the boxfish studied were very near the origin (that is, almost no lift at zero angle of attack). The difference might be error due to the fact that the actual boxfish dimensions were not reproduced in the solid model.

Third, it was found that vortices formed along the eye ridges of all the boxfish: this was clear from the simulation - the eye ridge regions for both the models have shown concentrated vorticity, which once again can be attributed to the sudden increase in the cross section profile. Finally, when both boxfish were positioned at various yaw angles, regions of stronger concentrated vorticity formed in far-field locations of the carapace when compared with nearfield areas, and vortex circulation was greatest in the posterior of center of mass, just as described in (Bartol, et. al, 2005). The vorticity contours for the yaw case are not presented here for the sake of brevity and can be found in (Kodati, 2006).

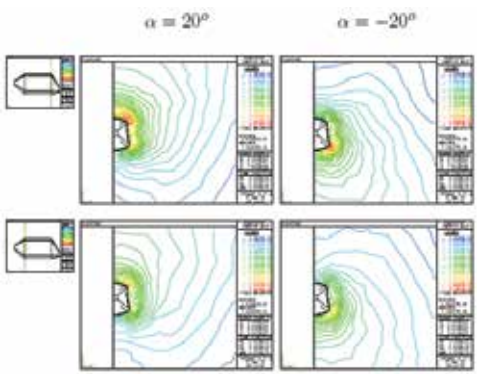


Fig. 10. Vorticity countour maps at various cross sections of the spotted boxfish.

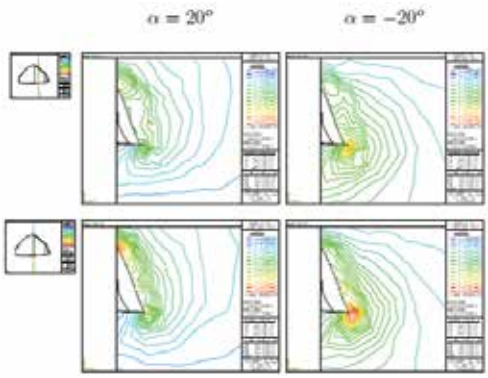


Fig. 11. Vorticity countour maps at various cross sections of the buffalo trunkfish.

From the analysis of the actual boxfish like shapes, it has been determined that the following two shape factors are mainly responsible for altering the vortex generating characteristics. First, change in the cross section profile: along the body for pitch stability and across the body for yaw stability. Second, sharpness of the keels that, in turn, depend on the concavity and convexity of the joining faces. These two characteristics were explored in the design of the MUV. Fig. 19 shows parameters used to change the body shape of the MUV. In order to verify the first of the two factors (cross section profile change), the shapes shown in Fig. 20 were considered. Both shapes were modified in the front region, and the rear change of area (the 'bump') was increased in the second shape. When flow was simulated at different pitch angles on both models, the peak vorticity was found to be higher for the second case at all angles of attack, as expected. The concavity of the side surfaces determines the sharpness of the keel and hence can affect the flow separation. To verify this, models with concave, flat, and convex surfaces were considered. There was not much variation in peak vorticity in the simulations for different pitch angles, but the vorticity was concentrated more heavily on the keels of the concave shape than on the flat and convex shapes. For yaw angles of attack, the peak vorticity was higher for the concave design and the vorticity was still concentrated near the keels. A comprehensive set of simulation results can be found in (Kodati, 2006).

4. Robotic prototype development

Boxfish employ a total of five fins to maneuver effectively. Biologists have observed three main swimming gaits employed at three different speed levels (Gordon, et. al, 2000). The design issue is how many of these fins can be practically implemented in a small underwater vehicle and how many degrees of freedom are used for each fin motion. For effective planar maneuverability, the degrees of freedom have to be distributed around the body. The present design incorporates a single DOF flexible tail fin for propulsion and a pair of 2 DOF pectoral fins for steering (yawing) and diving (pitching). Although the dorsal and anal fins of boxfish are believed to play a role in generating low recoil movement (Gordon, et. al, 2000), they are not included in the design due to size constraints.

The 2DOF pectoral fins use flapping and rotational motion in the so-called rowing mode. Rowing mode is a drag based thrust generation stroke with a full cycle of the fins. The power stroke is a quick backward push of the oil with the chord length perpendicular to the water flow. The recovery stroke involves bringing back the fin with the chord length parallel

to the flow. The rotation DOF is used to change the orientation of the fin at stroke reversals. This type of system can generate substantial turning moments about the body center of mass for sharp turns (Walker, 2000). The pectoral fins can also be used effectively as lifting surfaces by holding them at a suitable angle of attack to an oncoming flow.

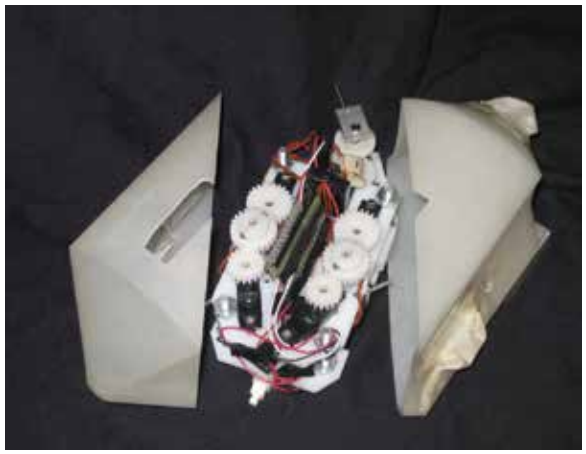


Fig. 12. Robotic prototype development.

A coaxial wrist mechanism has been designed that is similar to the robotic flapper design presented in (Kodati, 2007). The difference is that there is no deviation DOF and the gearbox size has been shrunk to a 2 cm × 2cm × 2cm volume using the smallest off-the-shelf miter gears available. The tail fin mechanism consists of a gear stage between the tail fin shaft and the motor shaft. The side fin flapping and rotation angles are coupled due to the no-slip condition of the wrist bevel gear box. A parallel plate structure was used to place the pectoral fin motors in front and above the wrist gear box. The electronic chips were placed on the top plate along with the pectoral fin motors and transmission mechanism. The battery pack, tail fin motor, and transmission were mounted on the bottom plate. The plates are made out of 0.125 inch thick Delrin sheet. A CNC mill was programmed to machine all the features on the plates.

Fig. 13. Swimming Experiments.

Accurate positioning of the fin flapping angles was required to control the force generation for turning and cruising. Servo motors with built in feedback were used in place of a DC motor and encoder combination. All the servos are daisy-chained to a servo controller that drives them to the position commanded by the processor. The onboard electronics include a Javelin-Stamp TMmicroprocessor module by Parallax Inc. and a serial motor controller PCB unit by Pololu Inc. The power supply for the motors, motor controller, and processor is provided by a pack of five 1.2V NiMH batteries. The microprocessor was programmed using an embedded Java version. A Java class routine was used to command the motor controller in serial communication mode using a built-in UART object.

The outer shape of the MUV was rapid prototyped using the stereolithograhy (3D printing) technique. Features for assembly and fin placement were incorporated into the shape. To seal the robot, layers of adhesive tape and oil resistant film were applied along the dividing line between the two body halves. The fin extensions were covered with latex sheet and attached to the shaft with a plastic O-ring. The latest configuration of the robot is one in which the body shape is assembled in diagonal halves as shown in Fig. 12. Fig. 13 shows the swimming prototype.

For the robot to be neutrally buoyant, the weight force must be equal to the buoyancy force. The buoyancy is determined by the volume of oil (experimental trials were conducted in a low viscosity, clear oil to avoid electrical shorting) displaced by the solid model. Copper bars, machined to fit underneath the chassis plates, were used to balance the buoyancy force and the weight force. The heaviest parts, such as the batteries and large copper plate, were designed to fit in the lower region of the robot. By placing the center of mass below the center of buoyancy, the robot was designed to have inherent stability. The location of the center of mass was found by hanging the prototype from a string attached at different points and taking an image. The images were overlaid on top of one another and a line was drawn to extend the line of the string. The approximate point of intersection of all the lines was the experimental center of mass.

The current robotic prototype can be programmed to use different gait styles. A CCD camera, by Allied Vision Technology, operating at 30 fps was used to record swimming trials and determine speed, recoil movement, and turning radius. The robot was run in the field of view of the camera and each image was saved in National Instrument's Vision Assistant for LabVIEW. A Virtual Instrument (VI) was created to calibrate the pixels to real world units of inches. Each image was then sequentially analyzed for the speed, recoil, and turning radius. The speed and turning radius were measured by the change in position of a point on the robot and a constant point ing the field of view. The average speed obtained was 0.0411 m/s with an almost zero turning radius. The recoil can be measured by the deviation in each frame from the straight line between the beginning and end points. The average recoil was found to be 0.826 cm. The gait used for the test is shown in Fig. 27. The green color of the fins indicates the phase of the stroke where forward propulsive thrust is produced, and the red color indicates the phase where the force generated is either minimal (like the recovery stroke of the pectoral fin) or in the direction opposite to the motion of the robot (as in the case of the caudal fin). The gait in Fig. 27 is one where a constant forward propulsive force is applied to the body during most of the caudal fin cycle.

5. Conclusion

In this chapter present the research leading to the design of a biologically inspired robotic boxfish using multiple fins. The design attempts to achieve the maneuverability of a small scale, multiple fin underwater system like that of the boxfish, while also incorporating a body with a self correcting mechanism.

Experimental studies were conducted to characterize and optimize the flapping fin propulsion of the tail fin. Towards this a three DOF robotic flapper was designed to act as the flapping tail or side fin of the fish and a fixed beam based force sensor was designed to measure the instantaneous forces generated by the fin motion. Tail fin with optimal shape induced flexibility has been found. The hydrodynamic force generation of tail fin has been modeled using a combination of quasi steady lift generation and empirically found drag and added mass effects. Fluid flow simulations on 3D CAD models of boxfish like shapes were used to arrive at the outer shape of the MUV. A robotic prototype of the MUV was designed based on the above analysis. The propulsion and maneuvering of the MUV is achieved by tail fin and two 2DOF side fins.

One of the immediate goals is to use the prototype to evaluate the efficiency of various gait patterns for a given set of flow conditions. Sensors and command architecture will also be used in future generations to give the robot greater autonomy. Currently, new mechanical sealing techniques and more processing power are being incorporated into the next generation of the prototype to facilitate longer trial runs and effective control of the robot. Studying fin-fin and body-fin interactions can help modify the design of body shape and/or fin kinematics for optimal thrust production or even better maneuverability. One such problem is that of the interaction between the side fin and tail fin. On a small robot like the present MUV the proximity of the side fin and the tail fin is more and can lead to strong interactions between them. Such mechanisms can be investigated using multiple flappers and/or body shape in the tow tank.

6. References

Bandyopadhyay, P. (2005) Trends in biorobotic autonomous undersea vehicles," IEEE Journal Of Oceanic Engineering, vol. 30, no. 1, January 2005.

Barrett, D.; Triantafyllou, M.; Yue, D.; Grosenbaugh, M. & Wolfgang, M. (1999) Drag reduction in fish-like locomotion," *Journal of Fluid Mechanics*, vol. 392, pp. 183–212, 1999.

Bartol, I.; Gharib, M.; Webb, P.; Weihs, D. & Gordon, M. (2005) Body-induced vortical flows: a common mechanism for self-corrective trimming control in boxfishes," *Journal of Experimental Biology*, vol. 208, pp. 327–344, 2005.

Breder, C. (1926) The locomotion of fishes, *Zoologica*, vol. 4, pp. 159–256, 1926.

Colgate, J. & Lynch, K. (2004) Mechanics and control of swimming: A review, *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, July, 2004.

Combes S. & Daniel, T. (2003) Flexural stiffness in insect wings. II. spatial distribution and dynamic wing bending." *Journal of Experimental Biology*, vol. 206, no. 17, 2003.

Epstein, M.; Colgate, J. & MacIver, M. (2006) Generating thrust with a biologically-inspired robotic ribbon fin," in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Beijing, China, 2006.

Fish, F.; Lauder, G.; Mittal, R.; Techet, A.; Triantafyllou, M.; Walker, J. & Webb, P. (2003) Conceptual design for the construction of a biorobotic auv based on biological hydrodynamics. *In Proceedings of the 13th international symposium on Unmanned Untethered Submersible Technology*, 2003.

Floyd, J.; Keegan T. & Sitti, M. (2006) A novel water running robot inspired by basilisk lizards," in *Proc. of IEEE/RSJ Int. Conf. On Intelligent Robots and Systems (IROS)*, October 2006.

Gordon, M.; Hove, J.; Webb, P. & Weihs, D. (2000) Boxfishes as unusually well-controlled autonomous underwater vehicles," *Physiol. Biochem. Zool.*, vol. 74, no. 6, pp. 663–671, 2000.

Kato,B. & Wicaksono, N. (2000) Development of biology-inspired autonomous underwater vehicle bass iii with high maneuverability," in *Proceedings of the 2000 International Symposium on Underwater Technology*, pp. 84–89, 2000.

Kodati, P. (2006) Biomimetic micro underwater vehicle with ostraciiform locomotion: System design, analysis and experiments, Master's thesis, University of Delaware, Newark, DE, USA, August 2006.

Kodati P. & Deng, X. (2007) Experimental studies on the hydrodynamics of a robotic ostraciiform tail fin," in *Proc. of IEEE/RSJ Int. Conf. On Intelligent Robots and Systems (IROS)*, October 2006, pp. 5418–5423.

Lachat, D.; Crespi, A. & Ijspeert, A. (2006) Boxybot: a swimming and crawling fish robot controlled by a central pattern generator," in *Proceedings of The first IEEE / RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics,* 2006,

Licht, S.; Polidoro, V.; Flores, M.; Hover, S. & Triantafyllou, M. (2004) Design and projected performance of a flapping foil auv," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, 2004.

Liao, G.; Beal D. & Triantafyllou, M. (2003) Fish exploiting vortices decrease muscle activity," *Science*, vol. 302, pp. 1566–1569, 2003.

Lindsey, C. (1978) Form, function and locomotory habits in fish, *Fish Physiology Volume VII: Locomotion*, W. S. Hoar and D. J. Randall, Eds. New York: Academic, 1978, pp. 1–100, 1978.

MacIver, M.; Fontaine, E. & Burdick, J. (2004) Designing future underwater vehicles: Principles and mechanisms of the weakly electric fish," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, July 2004.

McIsaac K. & Ostrowski, J. (2003) Motion planning for anguilliform locomotion, *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, August 2003.

Mohseni, K. (2004) Zero-mass pulsatile jets for unmanned underwater vehicle maneuvering, in *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, Chicago, IL, September 2004.

Sfakiotakis, M.; Lane, D. & Davies, J. (1999) Review of fish swimming modes for aquatic locomotion, *IEEE Journal of Oceanic Engineering*, vol. 24, no. 2, April 1999.

Triantafyllou, M. & Triantafyllou, G. (1995) An efficient swimming machine, *Scientific American*, vol. 272, no. 3, 1995.

Walker, J. (2000) Does a rigid body limit manuerability?, *Journal of Experimental Biology*, vol. 203, pp. 3391–3396, 2000.

Yuh, J. (2000) Design and control of autonomous underwater robots: A survey," *Autonomous Robots*, vol. 8, pp. 7–24, 2000.

Zhu, Q.; Wolfgang, M.; Yue, D. & Triantafyllou, M. (2002) Threedimensional flow structures and vorticity control in fish-like swimming, *Journal of Fluid Mechanics*, vol. 468, pp. 1–28, October 2002.

7

A Survey of Control Allocation Methods for Underwater Vehicles

Thor I. Fossen[1,2], Tor Arne Johansen[1] and Tristan Perez[3]
[1]*Dept. of Eng. Cybernetics, Norwegian Univ. of Science and Techn.*
[2]*Centre for Ships and Ocean Structures, Norwegian Univ. of Science and Techn.*
[3]*Centre of Excellence for Complex Dyn. Syst. and Control, Univ. of Newcastle,*
[1,2]*Norway*
[3]*Australia*

1. Introduction

A *control allocation system* implements a function that maps the desired control forces generated by the vehicle motion controller into the commands of the different actuators. In order to achieve high reliability with respect to sensor failure, most underwater vehicles have more force-producing actuators than the necessary number required for nominal operations. Therefore, it is common to consider the motion control problem in terms of generalised forces—independent forces affecting the different degrees of freedom—, and use a control allocation system. Then, for example, in case of an actuator failure the remaining ones can be reconfigured by the control allocation system without having to change the motion controller structure and tuning.

The control allocation function hardly ever has a close form solution; instead the values of the actuator commands are obtained by solving a constrained optimization problem at each sampling period of the digital motion control implementation loop. The optimization problem aims at producing the demanded generalized forces while at the same time minimizing the use of control effort (power).

Control allocation problems for underwater vehicles can be formulated as optimization problems, where the objective typically is to produce the specified generalized forces while minimizing the use of control effort (or power) subject to actuator rate and position constraints, power constraints as well as other operational constraints. In addition, singularity avoidance for vessels with rotatable thrusters represents a challenging problem since a non-convex nonlinear program must be solved. This is useful to avoid temporarily loss of controllability. In this article, a survey of control allocation methods for over-actuated underwater vehicles is presented. The methods are applicable for both surface vessels and underwater vehicles.

Over-actuated control allocation problems are naturally formulated as optimization problems as one usually wants to take advantage of all available degrees of freedom (DOF) in order to minimize power consumption, drag, tear/wear and other costs related to the use of control, subject to constraints such as actuator position limitations, e.g. Enns (1998), Bodson (2002) and Durham (1993). In general, this leads to a constrained optimization

problem that is hard to solve using state-of-the-art iterative numerical optimization software at a high sampling rate in a safety-critical real-time system with limiting processing capacity and high demands for software reliability. Still, real-time iterative optimization solutions can be used; see Lindfors (1993), Webster and Sousa (1999), Bodson (2002), Harkegård (2002) and Johansen, Fossen, Berge (2004). Explicit solutions can also be found and implemented efficiently by combining simple matrix computations, logic and filtering; see Sørdalen (1997), Berge and Fossen (1997) and Lindegaard and Fossen (2003).

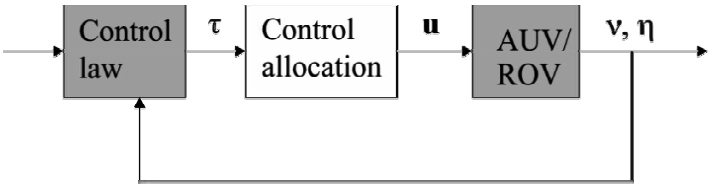


Fig. 1. Block diagram illustrating the control allocation problem.

The paper presents a survey of control allocation methods with focus on mathematical representation and solvability of thruster allocation problems. The paper is useful for university students and engineers who want to get an overview of state-of-the art control allocation methods as well as advance methods to solve more complex problems.

1.1 Problem formulation

Consider an underwater vehicle (Fossen, 2002):

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \\ \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) &= \boldsymbol{\tau} \end{aligned} \tag{1.1}$$

that is controlled by designing a feedback control law of *generalized control forces*:

$$\boldsymbol{\tau} = \mathbf{B}(\boldsymbol{\alpha})\mathbf{u} \in \mathbf{R}^n \tag{1.2}$$

where $\boldsymbol{\alpha} \in \mathbf{R}^p$ is a vector azimuth angles and $\mathbf{u} \in \mathbf{R}^r$ are *actuator commands*. For marine vehicles, some control forces can be rotated an angle about the *z*-axis and produce force components in the *x*- and *y*-directions, or about the *y*-axis and produce force components in the *x*- and *z*-directions. This gives additional control inputs $\boldsymbol{\alpha}$ which must be computed by the control allocation algorithm. The control law uses feedback from position/attitude $\boldsymbol{\eta} = [x, y, z, \varphi, \theta, \psi]^T$ and velocity $\boldsymbol{\nu} = [u, v, w, p, q, r]^T$ as shown in Figure 1.

For marine vessels with controlled motion in *n* DOF it is necessary to distribute the generalized control forces $\boldsymbol{\tau}$ to the actuators in terms of control inputs $\boldsymbol{\alpha}$ and $\mathbf{u}$. Consider (1.2) where $\mathbf{B}(\boldsymbol{\alpha}) \in \mathbf{R}^{n \times r}$ is the input matrix. If B has full rank (equal to n) and $r > n$, you have control forces in all relevant directions, this is an *over-actuated control* problem. Similarly, the case $r < n$ is referred to as an *under-actuated control* problem.

Computation of $\boldsymbol{\alpha}$ and $\mathbf{u}$ from $\boldsymbol{\tau}$ is a model-based optimization problem which in its simplest form is unconstrained while physical limitations like input amplitude and rate saturations imply that a constrained optimization problem must be solved. Another complication is actuators that can be rotated at the same time as they produce control forces. This increases the number of available controls from *r* to *r*+*p*.

2. Actuator models

The control force due to a propeller, a rudder, or a fin can be written

$$F = ku \tag{1.3}$$

where k is the force coefficient and u is the control input depending on the actuator considered; see Table 1. The linear model $F=ku$ can also be used to describe nonlinear monotonic control forces. For instance, if the rudder force F is quadratic in rudder angle δ, that is

$$F = k\delta|\delta|, \tag{1.4}$$

the choice $u = \delta|\delta|$, which has a unique inverse $\delta = sign(u)\sqrt{|u|}$, satisfies (1.3).

Actuator	u	α	$\mathbf{f}^T$
Main propeller/longitudinal thrusters	pitch/rpm	-	$[F_x,0,0]$
Transverse thrusters	pitch/rpm	-	$[0,F_y,0]$
Rotatable thruster in the horizontal plane	pitch/rpm	angle	$[F_x\cos\alpha, F_x\sin\alpha, 0]$
Rotatable thruster in the vertical plane	pitch/rpm	angle	$[F_z\sin\alpha, 0, F_z\cos\alpha]$
Aft rudders	angle	-	$[0,F_y,0]$
Stabilizing fins	angle	-	$[0,0,F_z]$

Table 1. Example of actuators and control variables.

For underwater vehicles the most common actuators are:

- **Main propellers/longitudinal thrusters** are mounted aft of the hull usually in conjunction with rudders. They produce the necessary force in the x-direction needed for transit.
- **Transverse thrusters** are sometime going through the hull of the vessel (tunnel thrusters). The propeller unit is then mounted inside a transverse tube and it produces a force in the y -direction. Tunnel thrusters are only effective at low speed which limits their use to low-speed maneuvering and DP.
- **Rotatable (azimuth) thrusters in the horizontal and vertical planes** are thruster units that can be rotated an angle α about the z-axis or y-axis to produce two force components in the horizontal or vertical planes, respectively. Azimuth thrusters are attractive in low-speed maneuvering and DP systems since they can produce forces in different directions leading to an over-actuated control problem that can be optimized with respect to power and possible failure situations.
- **Aft rudders** are the primary steering device for conventional vessels. They are located aft of the vessel and the rudder force F_y will be a function of the rudder deflection (the

drag force in the x-direction is usually neglected in the control analysis). A rudder force in the y-direction will produce a yaw moment which can be used for steering control.

- **Stabilizing fins** are used for damping of vertical vibrations and roll motions. They produce a force F_z in the z-directions which is a function of the fin deflection. For small angles this relationship is linear. Fin stabilizers can be retractable allowing for selective use in bad weather. The lift forces are small at low speed so the most effective operating condition is in transit.
- **Control surfaces** can be mounted at different locations to produce lift and drag forces. For underwater vehicles these could be fins for diving, rolling, and pitching, rudders for steering, etc.

Table 1 implies that the forces and moments in 6 DOF due to the force vector $\mathbf{f} = [F_x, F_y, F_z]^T$ can be written

$$\boldsymbol{\tau} = \begin{bmatrix} \mathbf{f} \\ \mathbf{r} \times \mathbf{f} \end{bmatrix} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ F_z l_y - F_y l_z \\ F_x l_z - F_z l_x \\ F_y l_x - F_x l_y \end{bmatrix} \tag{1.5}$$

where $\mathbf{r} = [l_x, l_y, l_z]^T$ are the moment arms. For azimuth thrusters in the horizontal plane the control force F will be a function of the rotation angle. Consequently, an azimuth thruster will have two force components $F_x = F\cos\alpha$ and $F_y = F\sin\alpha$, while the main propeller aft of the vehicle only produces a longitudinal force $F_x = F$, see Table 1.

2.1 Thrust configuration matrix for non-rotatable actuators

The control forces and moments for the fixed thruster case (no rotatable thrusters) can be written

$$\boldsymbol{\tau} = \mathbf{T}\mathbf{f} \tag{1.6}$$

where $\mathbf{T} \in \mathbb{R}^{n \times r}$ is the *thrust configuration matrix*. The control forces satisfies,

$$\mathbf{f} = \mathbf{K}\mathbf{u}, \tag{1.7}$$

with control inputs $\mathbf{u} = [u_1, ..., u_r]^T$. The *force coefficient matrix* $\mathbf{K} \in \mathbb{R}^{r \times r}$ is diagonal,

$$\mathbf{K} = diag\{k_1, ..., k_r\}. \tag{1.8}$$

The actuator configuration matrix is defined in terms of a set of column vectors $\mathbf{t}_i \in \mathbb{R}^n$ according to

$$\mathbf{T}(\boldsymbol{\alpha}) = [\mathbf{t}_1, ..., \mathbf{t}_r]. \tag{1.9}$$

If we consider 6 DOF motions, the columns vectors can be derived from (1.5) and (1.9) according to

$$\mathbf{t}_i = \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ l_{z_i} \\ l_{y_i} \end{bmatrix}}_{\text{main propeller}} \quad \mathbf{t}_i = \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ -l_{z_i} \\ 0 \\ l_{x_i} \end{bmatrix}}_{\text{tunnel thruster and aft rudder}} \quad \mathbf{t}_i = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ l_{y_i} \\ -l_x \\ 0 \end{bmatrix}}_{\text{stabilizing fin}} \tag{1.10}$$

2.2 Thrust configuration matrix for rotatable actuators

A more general representation of (1.6) is,

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{T}(\boldsymbol{\alpha})\mathbf{f} \\ &= \mathbf{T}(\boldsymbol{\alpha})\mathbf{K}\mathbf{u}, \end{aligned} \tag{1.11}$$

where the thrust configuration matrix $\mathbf{T}(\boldsymbol{\alpha}) \in \mathbb{R}^{n \times r}$ varies with the azimuth angles

$$\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_p]^T. \tag{1.12}$$

The azimuth thruster in the horizontal plane are defined in terms of the column vector

$$\mathbf{t}_i = \underbrace{\begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \\ 0 \\ -l_{z_i}\sin\alpha_i \\ l_{z_i}\sin\alpha_i \\ l_{x_i}\sin\alpha_i - l_{y_i}\cos\alpha_i \end{bmatrix}}_{\text{azimuth thruster in the horizontal plane}}, \quad \mathbf{t}_i = \underbrace{\begin{bmatrix} \sin\alpha_i \\ 0 \\ \cos\alpha_i \\ l_{y_i}\cos\alpha_i \\ l_{z_i}\sin\alpha_i - l_{x_i}\cos\alpha_i \\ -l_{y_i}\sin\alpha_i \end{bmatrix}}_{\text{azimuth thruster in the vertical plane}} \tag{1.13}$$

where the coordinates $(l_{x_i}, l_{y_i}, l_{z_i})$ denotes the location of the actuator with respect the body fixed coordinate system. Similar expressions can be derived for thrusters that are rotatable about the *x*- and *y*-axes.

2.3 Extended thrust configuration matrix for rotatable actuators

When solving the control allocation optimization problem an alternative representation to (1.10) is attractive to use. Equation (1.11) is nonlinear in the controls $\boldsymbol{\alpha}$ and $\boldsymbol{u}$. This implies that a nonlinear optimization problem must be solved. In order to avoid this, the rotatable thrusters can be treated as two forces.

Consider a rotatable thruster in the horizontal plane (the same methodology can be used for thrusters that can be rotated in the vertical plane),

$$\begin{aligned} F_{x_i} &= F_i \cos\alpha_i \\ &= k_i u_i \cos\alpha_i, \end{aligned} \tag{1.14}$$

$$\begin{aligned} F_{y_i} &= F_i \sin\alpha_i \\ &= k_i u_i \sin\alpha_i. \end{aligned} \tag{1.15}$$

Next, we define an *extended force* vector according to

$$\mathbf{f}_e = \mathbf{K}_e \mathbf{u}_e \tag{1.16}$$

such that

$$\boldsymbol{\tau} = \mathbf{T}_e \mathbf{K}_e \mathbf{u}_e \tag{1.17}$$

where $\mathbf{T}_e$ and $\mathbf{K}_e$ are the extended thrust configuration and thrust coefficient matrices, respectively and $\mathbf{u}_e$ is a vector of extended control inputs where the azimuth controls are modelled as

$$\begin{aligned} u_{ix} &= u_i \cos\alpha_i \\ u_{iy} &= u_i \sin\alpha_i \end{aligned} \tag{1.18}$$

The following examples show how this model can be established for an underwater vehicle equipped with two main propellers and two azimuth thrusters in the horizontal plane.

Example 1: Thrust configuration matrices for an ROV/AUV with rotatable thrusters

The horizontal plane forces X and Y in surge and sway, respectively and the yaw moment N satisfy (see Figure 2),

$$\boldsymbol{\tau} = \mathbf{T}(\boldsymbol{\alpha})\mathbf{K}\mathbf{u} \tag{1.19}$$

$$\Updownarrow$$

$$\begin{bmatrix} X \\ Y \\ N \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ l_{x_1}\sin\alpha_1 - l_{y_1}\cos\alpha_1 & l_{x_2}\sin\alpha_2 - l_{y_2}\cos\alpha_2 & l_{y_3} & l_{y_4} \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 & 0 \\ 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 \\ 0 & 0 & 0 & k_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}. \tag{1.20}$$

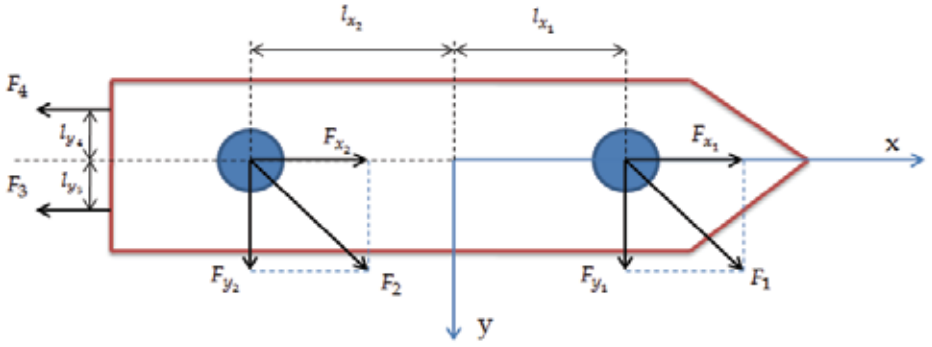


Fig. 2. ROV/AUV equipped with two azimuth thrusters (forces F_1 and F_2) and two main propellers (forces F_3 and F_4). The azimuth forces are decomposed along the x- and y-axis.

By using the extended thrust vector, (1.19) can be rewritten as,

$$\boldsymbol{\tau} = \mathbf{T}_e \mathbf{K}_e \mathbf{u}_e \tag{1.21}$$
$$\Updownarrow$$

$$\begin{bmatrix} X \\ Y \\ N \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & l_{x_1} & 0 & -l_{x_2} & l_{y_3} & l_{y_4} \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & k_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & k_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & k_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & k_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & k_4 \end{bmatrix} \begin{bmatrix} u_{1x} \\ u_{1y} \\ u_{2x} \\ u_{2y} \\ u_3 \\ u_4 \end{bmatrix}. \tag{1.22}$$

Notice that $\mathbf{T}_e$ *is constant while* $\mathbf{T}(\boldsymbol{\alpha})$ *depends on* $\boldsymbol{\alpha}$*. This means that the extended control input vector* $\mathbf{u}_e$ *can be solved directly from (1.21) by using a pseudo-inverse. This is not the case for (1.20) which represents a nonlinear optimization problem. The azimuth controls can then be derived from the extended control vector* $\mathbf{u}_e$ *by mapping the pairs* (u_{1x}, u_{1y}) *and* (u_{2x}, u_{2y}) *using the relations,*

$$\begin{aligned} u_1 &= \sqrt{u_{1x}^2 + u_{1y}^2}, \quad \alpha_1 = \operatorname{atan2}(u_{1y}, u_{1x}), \\ u_2 &= \sqrt{u_{2x}^2 + u_{2y}^2}, \quad \alpha_2 = \operatorname{atan2}(u_{2y}, u_{2x}). \end{aligned} \tag{1.23}$$

The last two controls u_3 *and* u_4 *are elements in* $\mathbf{u}_e$*.* □

3. Linear quadratic unconstrained control allocation

The simplest allocation problem is the one where all control forces are produced by thrusters in fixed directions alone or in combination with rudders and control surfaces such that

$$\boldsymbol{\alpha} = \text{constant}, \qquad \mathbf{T} = \mathbf{T}(\boldsymbol{\alpha}) = \text{constant}.$$

Assume that the allocation problem is *unconstrained*-i.e., there are no bounds on the vector elements f_i, α_i and u_i and their time derivatives. Saturating control and constrained control allocation are discussed in Sections 4-5.

For marine craft where the configuration matrix $\boldsymbol{T}$ is square or non-square $(r \geq n)$, that is there are equal or more control inputs than controllable DOF, it is possible to find an optimal distribution of control forces **f**, for each DOF by using an explicit method. Consider the unconstrained least-squares (LS) optimization problem (Fossen & Sagatun, 1991),

$$\begin{aligned} &\min_{\mathbf{f}} \ \left\{ J = \mathbf{f}^T \mathbf{W} \mathbf{f} \right\} \\ &\text{subject to: } \boldsymbol{\tau} - \mathbf{T}\mathbf{f} = \mathbf{0}. \end{aligned} \tag{1.24}$$

Here **W** is a positive definite matrix, usually diagonal, weighting the control forces. For marine craft which have both control surfaces and propellers, the elements in **W** should be

selected such that using the control surfaces is much more inexpensive than using the propellers.

3.1 Explicit solution for α = constant using lagrange multipliers

Define the *Lagrangian* (Fossen, 2002),

$$L(\mathbf{f},\boldsymbol{\lambda}) = \mathbf{f}^T\mathbf{W}\mathbf{f} + \boldsymbol{\lambda}^T(\boldsymbol{\tau} - \mathbf{T}\mathbf{f}), \tag{1.25}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^r$ is a vector of Lagrange multipliers. Consequently, differentiating the Lagrangian L with respect to $\mathbf{f}$, yields

$$\frac{\partial L}{\partial \mathbf{f}} = 2\mathbf{W}\mathbf{f} - \mathbf{T}^T\boldsymbol{\lambda} = \mathbf{0} \quad \Rightarrow \quad \mathbf{f} = \frac{1}{2}\mathbf{W}^{-1}\mathbf{T}^T\boldsymbol{\lambda}. \tag{1.26}$$

Next, assume that $\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^T$ is non-singular such that

$$\boldsymbol{\tau} = \mathbf{T}\mathbf{f} = \frac{1}{2}\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^T\boldsymbol{\lambda} \Rightarrow \boldsymbol{\lambda} = 2(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^T)^{-1}\boldsymbol{\tau}. \tag{1.27}$$

This gives

$$\boldsymbol{\lambda} = 2(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^T)^{-1}\boldsymbol{\tau}, \tag{1.28}$$

Substituting (1.28) into (1.27) yields,

$$\mathbf{f} = \mathbf{T}_w^{\dagger}\boldsymbol{\tau}, \qquad \mathbf{T}_w^{\dagger} = \mathbf{W}^{-1}\mathbf{T}^T(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^T)^{-1}, \tag{1.29}$$

where $\mathbf{T}_w^{\dagger}$ is recognized as the *generalized inverse*. For the case ***W=I***, that is equally weighted control forces, (1.29) reduces to the *Moore-Penrose pseudo inverse,*

$$\mathbf{T}^{\dagger} = \mathbf{T}^T(\mathbf{T}\mathbf{T}^T)^{-1}. \tag{1.30}$$

Since $\mathbf{f} = \mathbf{T}_w^{\dagger}\boldsymbol{\tau}$, the control input vector ***u*** can be computed from (1.7) as,

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{T}_w^{\dagger}\boldsymbol{\tau}. \tag{1.31}$$

Notice that this solution is valid for all α but not optimal with respect to a time-varying α.

3.2 Explicit solution for varying α using Lagrange multipliers

In the unconstraint case a time-varying α can be handled by using an extended thrust representation similar to Sørdalen (1997). Consider the ROV/AUV model in Example 1 where,

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{T}_e\mathbf{f}_e \\ &= \mathbf{T}_e\mathbf{K}_e\mathbf{u}_e \end{aligned} \tag{1.32}$$

Application of (1.29) now gives,

$$\begin{aligned} \mathbf{f}_e &= \mathbf{T}_w^{\dagger}\boldsymbol{\tau} \\ \mathbf{u}_e &= \mathbf{K}_e^{-1}\mathbf{f}_e, \end{aligned} \tag{1.33}$$

where $\mathbf{u}_e = [u_{1x}, u_{1y}, u_{2x}, u_{2y}, u_3, u_4]^T$ and $\mathbf{f}_e = [f_1, f_2, f_3, f_4, f_5, f_6]^T$. The optimal azimuth angles and thrust commands are then found as

$$\begin{aligned} u_1 &= \sqrt{u_{1x}^2 + u_{1y}^2} = \frac{1}{k_1}\sqrt{f_1^2 + f_2^2}, \quad \alpha_1 = \operatorname{atan2}(u_{1y}, u_{1x}), \\ u_2 &= \sqrt{u_{2x}^2 + u_{2y}^2} = \frac{1}{k_2}\sqrt{f_3^2 + f_4^2}, \quad \alpha_2 = \operatorname{atan2}(u_{2y}, u_{2x}), \\ u_3 &= \frac{f_5}{k_3}, \\ u_4 &= \frac{f_6}{k_4}. \end{aligned} \tag{1.34}$$

The main problem is that the optimal solution for α_1 and α_2 can jump at each sample which requires proper filtering. In the next sections, we propose other solutions to this problem.

4. Linear quadratic constrained control allocation

In practical systems it is important to minimize the power consumption by taking advantage of the additional control forces in an over-actuated control problem. It is also important to take into account actuator limitations like saturation, tear and wear as well as other constraints such as forbidden sectors, and overload of the power system. In general this leads to a *constrained* optimization problem.

4.1 Explicit solution for α = constant using piecewise linear functions (non-rotatable actuators)

An explicit solution approach for parametric quadratic programming has been developed by Tøndel *et al.* (2003) while applications to marine vessels are presented by Johansen *et al.* (2005). In this work the constrained optimization problem is formulated as

$$\begin{aligned} \min_{\mathbf{f},\mathbf{s},\bar{f}} \left\{ J = \mathbf{f}^T\mathbf{W}\mathbf{f} + \mathbf{s}^T\mathbf{Q}\mathbf{s} + \beta\bar{f} \right\} \\ \text{subject to:} \\ \mathbf{T}\mathbf{f} = \boldsymbol{\tau} + \mathbf{s} \\ \mathbf{f}_{\min} \le \mathbf{f} \le \mathbf{f}_{\max} \\ -\bar{f} \le f_1, f_2, \ldots f_r \le \bar{f} \end{aligned} \tag{1.35}$$

where $\mathbf{s} \in \mathbb{R}^n$ is a vector of *slack variables* and forces

$$\mathbf{f} = [f_1, f_2, \ldots, f_r]^T \in \mathrm{R}^r \tag{1.36}$$

The first term of the criterion corresponds to the LS criterion (1.25), while the third term is introduced to minimize the largest force $\overline{f} = \max_i \mid f_i \mid$ among the actuators. The constant $\beta \geq 0$ controls the relative weighting of the two criteria. This formulation ensures that the constraints $f_i^{\min} \leq f_i \leq f_i^{\max} \quad (i = 1,\ldots,r)$ are satisfied, if necessary by allowing the resulting generalized force $\mathbf{Tf}$ to deviate from its specification $\boldsymbol{\tau}$. To achieve accurate generalized force, the slack variable should be close to zero. This is obtained by choosing the weighting matrix $\mathbf{Q} \gg \mathbf{W} > 0$. Moreover, saturation and other constraints are handled in an optimal manner by minimizing the combined criterion (1.35). Let

$$\mathbf{p} = [\boldsymbol{\tau}^T, \mathbf{f}_{\min}^T, \mathbf{f}_{\max}^T, \beta]^T \in \mathrm{R}^{n+2r+1}, \tag{1.37}$$

denote the parameter vector and,

$$\mathbf{z} = [\mathbf{f}^T, \mathbf{s}^T, \overline{f}]^T \in \mathrm{R}^{r+n+1}. \tag{1.38}$$

Hence, it is straightforward to see that the optimization problem (1.35) can be reformulated as a QP problem:

$$\begin{aligned} &\min_{\mathbf{z}} \left\{ J = \mathbf{z}^T \boldsymbol{\Phi} \mathbf{z} + \mathbf{z}^T \mathbf{R} \mathbf{p} \right\} \\ &\text{subject to:} \\ &\mathbf{A}_1 \mathbf{z} = \mathbf{C}_1 \mathbf{p} \\ &\mathbf{A}_2 \mathbf{z} \leq \mathbf{C}_2 \mathbf{p} \end{aligned} \tag{1.39}$$

where:

$$\boldsymbol{\Phi} := \begin{bmatrix} \mathbf{W} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{n\times r} & \mathbf{Q} & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{1\times r} & \mathbf{0}_{1\times n} & 0 \end{bmatrix}, \quad \mathbf{R} := \begin{bmatrix} \mathbf{0}_{(r+n+1)\times(n+2r)} & \begin{bmatrix} \mathbf{0}_{(r+n)\times 1} \\ 1 \end{bmatrix} \end{bmatrix}$$

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{T} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times 1} \end{bmatrix}, , \quad \mathbf{A}_2 := \begin{bmatrix} -\mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & -\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \end{bmatrix} \tag{1.40}$$

$$\mathbf{C}_1 = \begin{bmatrix} \mathbf{I}_{n\times n} & \mathbf{0}_{n\times(2r+1)} \end{bmatrix}, , \quad \mathbf{C}_2 := \begin{bmatrix} \mathbf{0}_{r\times n} & -\mathbf{I}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{I}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \end{bmatrix}$$

Since $\mathbf{W} > 0$ and $\mathbf{Q} > 0$ this is a convex quadratic program in $\mathbf{z}$ parameterized by $\mathbf{p}$. Convexity guarantees that a global solution can be found. The optimal solution $\mathbf{z}^*(\mathbf{p})$ is a continuous piecewise linear function $\mathbf{z}^*(\mathbf{p})$ defined on any subset,

$$\mathbf{p}_{\min} \leq \mathbf{p} \leq \mathbf{p}_{\max} \tag{1.41}$$

of the parameter space. Moreover, an exact representation of this piecewise linear function can be computed off-line using multi-parametric QP algorithms (Tøndel and Johansen, 2003b) or the Matlab *Multi-Parametric Toolbox (MPT)* by Kvasnica, Grieder and Baotic (2004). Consequently, it is not necessary to solve the QP (1.36) in real time for the current value of $\boldsymbol{\tau}$ and the parameters $f_{\min}, f_{\max}$ and β, if they are allowed to vary.
In fact it suffices to evaluate the known piecewise linear function $\mathbf{z}^*(\mathbf{p})$ as a function of the given parameter vector p which can be done efficient with a small amount of computations. For details on the implementation aspects of the mp-QP algorithm; see Johansen et al. (2003) and references therein. An on-line control allocation algorithm is presented in Tøndel et al. (2003a).

4.2 Explicit solution for varying α using piecewise linear functions (rotatable thrusters and rudders)

An extension of the mp-QP algorithm to marine vessels equipped with azimuthing thrusters and rudders has been given by Johansen et al. (2003). A propeller with a rudder can produce a thrust vector within a range of directions and magnitudes in the horizontal plane for low-speed maneuvering and dynamic positioning. The set of attainable thrust vectors is non-convex because significant lift can be produced by the rudder only with forward thrust. The attainable thrust region can, however, be decomposed into a finite union of convex polyhedral sets. A similar decomposition can be made for azimuthing thrusters including forbidden sectors. Hence, this can be formulated as a mixed-integer-like convex quadratic programming problem and by using arbitrarily number of rudders as well as thrusters and other propulsion devices can be handled. Actuator rate and position constraints are also taken into account. Using a multi-parametric quadratic programming software, an explicit piecewise linear representation of the least-squares optimal control allocation law can be pre-computed. The method is illustrated using a scale model of a supply vessel in a test basin, see Johansen et al. (2003) for details, and using a scale model of a floating platform in a test basin, see Spjøtvold (2008).

4.3 Explicit solutions based on minimum norm and null-space methods (non-rotatable actuators)

In flight and aerospace control systems, the problems of control allocation and saturating control have been addressed by Durham (1993, 1994a, 1994b). They also propose an explicit solution to avoid saturation referred to as the direct method. By noticing that there are infinite combinations of admissible controls that generate control forces on the boundary of the closed subset of attainable controls, the direct method calculates admissible controls in the interior of the attainable forces as scaled down versions of the unique solutions for force demands. Unfortunately it is not possible to minimize the norm of the control forces on the boundary or some other constraint since the solutions on the boundary are unique. The

computational complexity of the algorithm is proportional to the square of the number of controls, which can be problematic in real-time applications.
In Bordignon and Durham (1995) the null space interaction method is used to minimize the norm of the control vector when possible, and still access the attainable forces to overcome the drawbacks of the direct method. This method is also explicit but much more computational intensive. For instance 20 independent controls imply that up to 3.4 billon points have to be checked at each sample. In Durham (1999) a computationally simple and efficient method to obtain near-optimal solutions is described. The method is based on prior knowledge of the controls' effectiveness and limits such that pre-calculation of several generalized inverses can be done.

4.4 Iterative solutions

An alternative to the explicit solution could be to use an iterative solution to solve the QP problem (Sørdalen, 1997). The drawback with the iterative solution is that several iterations may have to be performed at each sample in order to find the optimal solution. The iterative approach is more flexibility for on-line reconfiguration, as for example a change in $\mathbf{W}$ may require that the explicit solutions are recalculated. Computational complexity is also greatly reduced by a warm start-i.e., the numerical solver is initialized with the solution of the optimization problem computed at the previous sample.
Finally, the offline computed complexity and memory requirements may be prohibited for the explicit solution to be applicable to large scale control allocation problems.

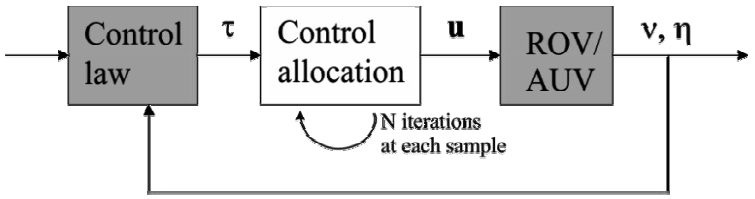


Fig. 3. Block diagram illustrating the iterative control allocation problem.

5. Nonlinear constrained control allocation (rotatable actuators)

The control allocation problem for vessels equipped with azimuth thrusters is in general a *non-convex* optimization problem that is hard to solve. The primary constraint is

$$\boldsymbol{\tau} = \mathbf{T}(\boldsymbol{\alpha})\mathbf{f}, \tag{1.42}$$

where $\boldsymbol{\alpha} \in \mathrm{R}^p$ denotes the azimuth angles. The azimuth angles must be computed at each sample together with the control inputs $\mathbf{u} \in \mathrm{R}^p$ which are subject to both amplitude and rate saturations. In addition, rotatable thrusters may only operate in feasible sectors $\alpha_{i,\min} \leq \alpha_i \leq \alpha_{i,\max}$ at a limiting turning rate $\dot{\boldsymbol{\alpha}}$. Another problem is that the inverse,

$$\mathbf{T}_w^T(\boldsymbol{\alpha}) = \mathbf{W}^{-1}\mathbf{T}^T(\boldsymbol{\alpha})[\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^T(\boldsymbol{\alpha})]^{-1}, \tag{1.43}$$

may not exist for certain $\boldsymbol{\alpha}$-values due to singularity. The consequence of such a singularity is that no force is produced in certain directions. This may greatly reduce dynamic performance and maneuverability as the azimuth angles can be changed slowly only. This suggests that the following criterion should be minimized (Johansen *et al.*, 2004),

$$\min_{\mathbf{f},\boldsymbol{\alpha},\mathbf{s}} \left\{ J = \sum_{i=1}^{r} \overline{P}_i \left| f_i \right|^{3/2} + \mathbf{s}^T \mathbf{Q} \mathbf{s} + (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \boldsymbol{\Omega} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + \frac{\rho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha}) \mathbf{W}^{-1} \mathbf{T}^{?}(\boldsymbol{\alpha}))} \right\} \tag{1.44}$$

subject to

$$\mathbf{T}(\boldsymbol{\alpha})\mathbf{f} = \boldsymbol{\tau} + \mathbf{s}$$

$$\mathbf{f}_{\min} \le \mathbf{f} \le \mathbf{f}_{\max}$$

$$\boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max}$$

$$\Delta\boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} - \boldsymbol{\alpha}_0 \le \Delta\boldsymbol{\alpha}_{\max}$$

where

- $\sum_{i=1}^{r} \overline{P}_i \left| f_i \right|^{3/2}$ represents power consumption where $\overline{P}_i > 0 \ (i = 1,\ldots,r)$ are positive weights.
- $\mathbf{s}^T \mathbf{Q} \mathbf{s}$ penalizes the error $\mathbf{s}$ between the commanded and achieved generalized force. This is necessary in order to guarantee that the optimization problem has a feasible solution for any $\boldsymbol{\tau}$ and $\boldsymbol{\alpha}_0$. The weight $\mathbf{Q} > 0$ is chosen so large that the optimal solution is $\mathbf{s} \approx \mathbf{0}$ whenever possible.
- $\mathbf{f}_{\min} \le \mathbf{f} \le \mathbf{f}_{\max}$ is used to limit the use of force (saturation handling).
- $\boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max}$ denotes the feasible sectors of the azimuth angles.
- $\Delta\boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} - \boldsymbol{\alpha}_0 \le \Delta\boldsymbol{\alpha}_{\max}$ ensures that the azimuth angles do not move to much within one sample taking $\boldsymbol{\alpha}_0$ equal to the angles at the previous sample. This is equivalent to limiting $|\dot{\boldsymbol{\alpha}}|$, -i.e. the turning rate of the thrusters.
- The term

$$\frac{\rho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha}) \mathbf{W}^{-1} \mathbf{T}^{T}(\boldsymbol{\alpha}))}$$

is introduced to avoid singular configurations given by $\det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^{T}(\boldsymbol{\alpha})) = 0$. To avoid division by zero, $\varepsilon > 0$, is chosen as a small number, while $\rho > 0$ is scalar weight. A large ρ ensures high maneuverability at the cost of higher power consumption and vice versa.

The optimization problem (1.44) is a non-convex nonlinear program and it requires a significant amount of computations at each sample (Nocedal and Wright, 1999). Consequently, the following two implementation strategies are attractive alternatives to nonlinear program efforts.

5.1 Dynamic solution using Lyapunov methods

In Johansen (2004) a control-Lyapunov approach has been used to develop an optimal dynamic control allocation algorithm. The proposed algorithm leads to asymptotic optimality. Consequently, the computational complexity compared to a direct nonlinear programming approach is considerably reduced. This is done by constructing the

optimizing control allocation algorithm as a dynamic update law which can be used together with a feedback control system. It is shown that the asymptotically optimal control allocation algorithm in interaction with an exponentially stable trajectory-tracking controller guarantees uniform boundedness and uniform global exponential convergence. A case study addressing low-speed maneuvering of an overactuated ship is used to demonstrate the performance of the control allocation algorithm. Extension to the adaptive case where thrust losses are estimated are given in (Tjønnås & Johansen, 2005), and extension to the case when actuator dynamics are considered explicitly in the control allocation is given in (Tjønnås & Johansen, 2007).

5.2 Iterative solutions using quadratic programming

The problem (1.42) can be locally approximated with a *convex* QP problem by assuming that:

1. the power consumption can be approximated by a quadratic term in $\mathbf{f}$, near the last force $\mathbf{f}_0$ such that $\mathbf{f} = \mathbf{f}_0 + \Delta\mathbf{f}$.
2. the singularity avoidance penalty can be approximated by a linear term linearized about the last azimuth angle $\boldsymbol{\alpha}_0$ such that $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \Delta\boldsymbol{\alpha}$.

The resulting QP criterion is (Johansen *et al.* , 2004):

$$\begin{aligned}
&\min_{\Delta\mathbf{f},\Delta\boldsymbol{\alpha},\mathbf{s}} \Big\{ J = (\mathbf{f}_0 + \Delta\mathbf{f})^T \mathbf{P}(\mathbf{f}_0 + \Delta\mathbf{f}) \\
&\quad + \mathbf{s}^T\mathbf{Q}\mathbf{s} + \Delta\boldsymbol{\alpha}^T\boldsymbol{\Omega}\Delta\boldsymbol{\alpha} \\
&\quad + \frac{\partial}{\partial\boldsymbol{\alpha}}\left(\frac{\rho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^T(\boldsymbol{\alpha}))} \right)\Big|_{\boldsymbol{\alpha}_0} \Delta\boldsymbol{\alpha} \Big\} \qquad (1.45) \\
&\text{subject to} \\
&\mathbf{s} + \mathbf{T}(\boldsymbol{\alpha}_0)\Delta\mathbf{f} + \tfrac{\partial}{\partial\boldsymbol{\alpha}}\big(\mathbf{T}(\boldsymbol{\alpha})\mathbf{f}\big)\big|_{\boldsymbol{\alpha}_0,\mathbf{f}_0} \Delta\boldsymbol{\alpha} = \boldsymbol{\tau} - \mathbf{T}(\boldsymbol{\alpha}_0)\mathbf{f}_0 \\
&\mathbf{f}_{\min} - \mathbf{f}_0 \le \mathbf{f} \le \mathbf{f}_{\max} - \mathbf{f}_0 \\
&\boldsymbol{\alpha}_{\min} - \boldsymbol{\alpha}_0 \le \Delta\boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max} - \boldsymbol{\alpha}_0 \\
&\Delta\boldsymbol{\alpha}_{\min} \le \Delta\boldsymbol{\alpha} \le \Delta\boldsymbol{\alpha}_{\max}
\end{aligned}$$

The convex QP problem (1.43) can be solved by using standard software for numerical optimization.

5.3 Iterative solutions using linear programming

Linear approximations to the thrust allocation problem have been discussed by Webster and Sousa (1999) and Lindfors (1993). In Linfors (1993) the azimuth thrust constraints

$$|f_i| = \sqrt{(f_i\cos\alpha_i)^2 + (f_i\sin\alpha_i)^2} \le f_i^{\max} \qquad (1.46)$$

are represented as circles in the $(f_i\cos\alpha_i, f_i\sin\alpha_i)$-plane. The nonlinear program is transformed to a linear programming (LP) problem by approximating the azimuth thrust constraints by straight lines forming a polygon. If 8 lines are used to approximate the circles (octagons), the worst case errors will be less than $\pm 4.0\%$. The criterion to be minimized is a linear combination of $|\mathbf{f}|$, that is magnitude of force in the *x*- and *y*-directions, weighted against the magnitudes

$$| \sqrt{(f_i \cos \alpha_i)^2 + (f_i \sin \alpha_i)^2} | \tag{1.47}$$

representing azimuth thrust. Hence, singularities and azimuth rate limitations are not weighted in the cost function. If these are important, the QP formulation should be used.

5.4 Explicit solution using the singular value decomposition and filtering techniques

An alternative method to solve the constrained control allocation problem is to use the singular value decomposition (SVD) and a filtering scheme to control the azimuth directions such that they are aligned with the direction where most force is required, paying attention to singularities (Sørdalen 1997). Results from sea trials have been presented in Sørdalen (1997). A similar technique using the damped-least squares algorithm has been reported in Berge and Fossen (1997) where the results are documented by controlling a scale model of a supply vessel equipped with four azimuth thrusters.

6. Case study: allocation problem formulation for an AUV with control surfaces

Some underwater vehicles perform all their missions at forward speed. In these applications, the vehicle hull design is streamlined so as to reduce hull drag, and the preferred type of control surface is the hydrofoil or fin. Hydrofoils produce lift, which is the useful force for controlling the motion of the vehicle. The side effect of lift generation, however, is drag—in other words, drag is the price we pay to obtain lift. Hence, for vehicles with several mounted control surfaces, the control allocation seeks the implementation of the demanded generalised forces while minimising the foil-induced drag. In this section, we formulate the control allocation problem for an AUV with two fixed thrusters and hydrofoil control surfaces.

Figure 4 shows INFANTE—an AUV built and operated by the Insituto Supetior Tecnico de Lisboa, Portugal. This AUV has two fixed thrusters at the stern, and six control surfaces: two horizontal fins mounted on the bow quarter, two horizontal fins mounted on the stern quarter, and two rudders mounted vertically behind the propellers.

Fig. 4. INFANTE-AUV. Picture courtesy of Dynamic Systems and Ocean Robotics Laboratory (DSOR), Instituto Superior Tecnico de Lisboa, Portugal. Copyright (c) 2001 DSOR-ISR.

Standard hydrofoil theory, see for example Marchaj (2000), establishes that the lift force produced by the hydrofoils is directed perpendicular to the incoming flow while the drag

force is directed along the incoming flow direction. The magnitude of the lift and drag forces can be modelled as,

$$L = \frac{1}{2}\rho_w A u_f^2 C_L \delta, \tag{1.48}$$

$$D = \frac{1}{2}\rho_w A u_f^2 C_D \delta^2, \tag{1.49}$$

where ρ_w is the water density, A is the area of the hydrofoil, u_f is the fluid velocity relative to the hydrofoil, C_L and C_D are the lift and drag coefficients respectively (measured experimentally), and δ is the angle of attack between the hydrofoil and the incoming flow.
Table 2 shows the different variables associated with the different control actuators considered in this case study. Notice that for the positive angle deflection of the control surfaces we use the right-hand rule along the direction of the rotation axis towards the tip.

Variable	Description	Positive convention
δ_{pb}	Port bow fin angle	Forward edge down
δ_{sb}	Starboard bow fin angle	Forward edge up
δ_{ps}	Port stern fin angle	Forward edge down
δ_{ss}	Starboard stern fin angle	Forward edge up
δ_{pr}	Port rudder angle	Forward edge to port
δ_{sr}	Starboard rudder angle	Forward edge to port
T_p	Port thuster thust	Forward
T_s	Starboard thuster thust	Forward

Table 2. Manipulated variables associated with the different actuators of the AUV shown in Figure 4.

For the control allocation problem, we will assume that the velocity u_f is either measured or estimated. We will also assume that the vehicle manoeuvres slowly from its equilibrium operational condition at forward speed. Hence, we can neglect the small drift angles; and thus, the lift and drag forces of the different hydrofoils can be considered to act along the x- and y-direction of the body-fixed coordinate system attached to the vessel. Furthermore, under the slow manoeuvring assumption and small drift angle, the angle of attack δ of the hydrofoils can be approximated by the mechanical angle of rotation of the hydrofoils.
For the particular vehicle under study, we can consider motion control objectives in 5DOF (surge, heave, pitch, roll, and yaw). With these objectives, the fins can be used to control heave, pitch and roll, the rudders to control yaw, and the thrusters to control surge. Then, we can simplify the allocation problem by taking a three-step approach:

1. Solve the allocation of the fins to obtain the deflection angles that implement the desired heave force and pitch and roll moments while minimising the induced drag.
2. Compute rudder angles based on the demanded yaw moment.
3. Compute thrust demand for the thrusters based on the demanded surge force while compensating for the fin and rudder induced drag forces.

The separation into these three steps simplifies the optimisation problem associated with the allocation. The first step results in a quadratic programme with linear constraints since only the lift forces are used. Then the rudders are used only for controlling the heading or yaw.

Finally, after computing the fin and rudder deflection angles, the thrust can be computed to implement the desired surge force and to compensate for the drag forces of the fins and rudders.

The above allocation scheme could be interpreted as a feed-forward compensation for the side effects of the fin and rudder drag induced forces.

Step 1: fin Allocation

Based on the above assumptions and the adopted positive convention for the variables shown in Table 1, we obtain the following vector of fin commands and force configuration matrix for heave, pitch and roll allocation

$$\mathbf{f}_{fins} = \begin{bmatrix} \delta_{pb} & \delta_{sb} & \delta_{ps} & \delta_{ss} \end{bmatrix}^T, \tag{1.50}$$

$$\mathbf{T}_{fins} = \begin{bmatrix} -k_L^b & k_L^b & -k_L^s & k_L^s \\ -x_{pb}\, k_L^b & x_{sb}\, k_L^b & -x_{ps}\, k_L^s & x_{ss}\, k_L^s \\ y_{pb} k_L^b & -y_{sb}\, k_L^b & y_{ps}\, k_L^s & -y_{ss}\, k_L^s \end{bmatrix}, \tag{1.51}$$

where

$$\begin{aligned} k_L^b &= \frac{1}{2} \rho_w A_b v_f^2 C_L^b, \\ k_L^s &= \frac{1}{2} \rho_w A_s v_f^2 C_L^s. \end{aligned} \tag{1.52}$$

Since the fin-induced drag is proportional to the square of the angle of attack, a natural objective function to minimize in the control allocation problem is a quadratic function. Depending on the difference in size and hydrodynamic characteristics of the bow and stern hydrofoils, we could perhaps use a different weighting to the two sets of fins. Thus, the fin allocation problem can be formulated as a standard quadratic program:

$$\begin{aligned} &\min_{\mathbf{f},\mathbf{s}} \left\{ \mathbf{f}^T \mathbf{W} \mathbf{f} + \mathbf{s}^T \mathbf{Q} \mathbf{s} \right\} \\ &\quad \text{subject to} \\ &\quad \mathbf{T}_{fins} \mathbf{f} = \mathbf{t} - \mathbf{s} \\ &\quad \mathbf{M} \mathbf{f} \le \mathbf{N} \\ &\quad \mathbf{s} \ge \mathbf{0}, \end{aligned} \tag{1.53}$$

with

$$\mathbf{W} = \begin{bmatrix} w_b & 0 & 0 & 0 \\ 0 & w_b & 0 & 0 \\ 0 & 0 & w_s & 0 \\ 0 & 0 & 0 & w_s \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} \mathbf{I}_{4\times 4} \\ -\mathbf{I}_{4\times 4} \end{bmatrix}, \quad \mathbf{N} = \begin{bmatrix} \Delta \\ \Delta \end{bmatrix}, \quad \Delta = \begin{bmatrix} \delta_{\max}^b \\ \delta_{\min}^b \\ \delta_{\max}^s \\ \delta_{\min}^s \end{bmatrix}, \tag{1.54}$$

where w_b and w_s represent the weighting for the bow and stern fins—note that only their relative value is of importance.

Step 2: Rudder Allocation
In nominal operational conditions, we can use the same deflection for both rudders. Hence, the allocation problem reduces to inverse of the mapping from angle to rudder moment:

$$\delta_{pr} = \delta_{sr} = \frac{N_c}{x_r \rho A_r v_{prop}^2 C_L^r}, \tag{1.55}$$

where x_r denotes the longitudinal position of the rudders relative to the adopted body-fixed reference system, v_{prop} is the flow velocity in the wake of the propeller, N_c is the yaw moment demanded by the vehicle motion controller.
Step 3: Thruster Allocation
In nominal operational conditions, we can use the same demand for the two thrusters. This demand is computed to implement the desired thrust demanded by the controller and to compensate the drag induced by the fins and rudders

$$T_p = T_s = \frac{1}{2}\left(X_c + X_{cs} \right), \tag{1.56}$$

where X_c is the surge force demanded by the vehicle motion controller, and X_{cs} is the added resistance due to the deflection of all the control surfaces

$$X_{cs} = k_D^b \left(\delta_{pb}^2 + \delta_{sb}^2 \right) + k_D^s \left(\delta_{ps}^2 + \delta_{ss}^2 \right) + k_D^r \left(\delta_{pr}^2 + \delta_{sr}^2 \right), \tag{1.57}$$

with the following drag related coefficients for the bow fins, stern fins, and rudders respectively

$$\begin{aligned} k_D^b &= \frac{1}{2} \rho_w A_b v_f^2 C_D^b, \\ k_D^s &= \frac{1}{2} \rho_w A_s v_f^2 C_D^s, \\ k_D^r &= \frac{1}{2} \rho_w A_r v_{prop}^2 C_D^r. \end{aligned} \tag{1.58}$$

In this section, we have considered a case study and formulated the control allocation problem for a particular AUV with two thrusters and six control surfaces. We have made some simplifying assumptions and considered the nominal operational conditions. Similar modelling procedures to that followed in this case study can be applied to other AUV with different actuators.

7. Conclusion

A survey of methods for control allocation of overactuated marine vessels has been presented. Both implicit and explicit methods formulated as optimization problems have been discussed. The objective has been to minimize the use of control effort (or power) subject to actuator rate and position constraints, power constraints as well as other operational constraints.
A case study of an AUV with control surfaces has been included in order to show how quadratic programming can be used to solve the control allocation problem.

8. References

Berge, S. P. and T. I. Fossen (1997). Robust Control Allocation of Overactuated Ships: Experiments With a Model Ship. *Proc. of the 4th IFAC Conference on Manoeuvring and Control of Marine Craft,* pp. 166-171, Brijuni, Croatia.

Bodson, M. (2002). Evaluation of Optimization Methods for Control Allocation. *Journal of Guidance, Control and Dynamics*, vol. 25, pp. 703-711.

Bordignon, K. A. and W. C. Durham (1995). Closed-Form Solutions to Constrained Control Allocation Problem. *Journal of Guidance, Control and Dynamics*, vol. 18, no. 5, pp. 1000-1007.

Durham, W. C. (1993). Constrained Control Allocation. *Journal of Guidance, Control and Dynamics*, vol. 16, no. 4, pp. 717-725.

Durham, W. C. (1994a). Constrained Control Allocation: Three Moment Problem. *Journal of Guidance, Control and Dynamics*, vol. 17, no. 2, pp. 330-336.

Durham, W. C. (1994b). Attainable Moments for the Constrained Control Allocation Problem. *Journal of Guidance, Control and Dynamics*, vol.17, no. 6, pp. 1371-1373.

Durham, W. C. (1999). Efficient, Near-Optimal Control Allocation. *Journal of Guidance, Control and Dynamics*, vol. 22, no. 2, pp. 369-372.

Enns, D. (1998). Control Allocation Approaches, *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit.* pp. 98-108, Reston, VA.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley and Sons Ltd., ISBN 0 471-94113-1.

Fossen, T. I. (2002). *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles,* Marine Cybernetics AS, ISBN 82-92356-00-2.

Fossen, T. I. and S. I. Sagatun (1991). Adaptive Control of Nonlinear Systems: A Case Study of Underwater Robotic Systems. *Journal of Robotic Systems*, vol. 8, no. 3, pp. 393-412.

Harkegård, O. (2002). Efficient Active Set Algorithms for Solving Constraint Least Squares Problems in Aircraft Control Allocation. *Proc. of the 41st IEEE Conference on Decision and Control (CDC'02)*, 2002.

Johansen, T. A. (2004). Optimizing Nonlinear Control Allocation. *Proc. of the IEEE Conf. Decision and Control (CDC'04)*, pp. 3435-3440, Nassau, Bahamas.

Johansen, T. A., T. I. Fossen and S. P. Berge (2004). Constraint Nonlinear Control Allocation with Singularity Avoidance using Sequential Quadratic Programming. *IEEE Transactions on Control Systems Technology*, vol. 12, pp. 211-216.

Johansen, T. A., T. I. Fossen and P. Tøndel (2005). Efficient Optimal Constrained Control Allocation via Multi-Parametric Programming. *AIAA Journal of Guidance, Control and Dynamics*}, vol. 28, pp. 506--515.

Johansen, T. A., T. P. Fuglseth, P. Tøndel and T. I. Fossen (2003). Optimal Constrained Control Allocation in Marine Surface Vessels with Rudders. *Proc. of the IFAC Conf. Manoeuvring and Control of Marine Craft*, Girona, Spain.

Kvasnica, M., P. Grieder and M. Baotic (2004). *Multi-Parametric Toolbox (MPT)*, <http://control.ee.ethz.ch/~mpt>.

Lindegaard, K.-P. and T. I. Fossen (2002). Fuel Efficient Control Allocation for Surface Vessels with Active Rudder Usage: Experiments with a Model Ship. *IEEE Transactions on Control Systems Technology,* vol. 11, pp. 850-862.

Lindfors, I. (1993). Thrust Allocation Method for the Dynamic Positioning System. *Proc. Of the 10th International Ship Control Systems Symposium (SCSS'93)*, pp. 3.93-3.106, Ottawa, Canada.

Marchaj, C. A. (2000). *Aero-hydrodynamic of Sailing*, 3rd Edition. Adlard Coles Publishing. ISBN 1-888671-18-1.

Nocedal, J. and S. J. Wright (1999). *Numerical Optimization.* Springer-Verlag, New York.

Sørdalen, O. J. (1997). Optimal Thrust Allocation for Marine Vessels. *Control Engineering Practice*, vol. 5, no. 9, pp. 1223-1231.

Spjøtvold, J. (2008). *Parametric Programming in Control Theory.* PhD thesis, Norwegian University of Science and Technology, Trondheim.

Tjønnås, J., T. A. Johansen (2005). Optimizing Nonlinear Adaptive Control Allocation. *Proc. of the IFAC World Congress*, Prague.

Tjønnås, J., T. A. Johansen (2007). Optimizing Adaptive Control Allocation with Actuator Dynamics. *Proc. of the IEEE Conference on Decision and Control*, New Orleans.

Tøndel, P., T. A. Johansen and A. Bemporad (2003a). An Algorithm for Multi-parametric Quadratic Programming and explicit MPC solutions. *Automatica*, vol. 39, pp. 489-497.

Tøndel, P., T. A. Johansen and A. Bemporad (2003b). Evaluation of Piecewise Affine Control via Binary Searh Tree. *Automatic*a, vol. 39, pp. 743-749.

Webster, W. C. and J. Sousa (1999). Optimum Allocation for Multiple Thrusters. *Proc. of the Int. Society of Offshore and Polar Engineers Conference (ISOPE'99),* Brest, France.

8

Making AUVs Truly Autonomous

Per Espen Hagen[1], Øyvind Hegrenæs[1], Bjørn Jalving[1], Øivind Midtgaard[2], Martin Wiig[2] and Ove Kent Hagen[2]
[1]*Kongsberg Maritime*
[2]*Norwegian Defence Research Establishment (FFI)*
Norway

1. Introduction

After decades of research and development, autonomous underwater vehicles (AUVs) are today becoming accepted by an increasing number of users in various military and civilian establishments. The number of AUV systems sold to civilian and military customers worldwide is well into triple digits. The bulk of these systems have been manufactured within the last five years, so the sector is in rapid growth.
AUVs provide a safe, cost-effective and reliable alternative to manned or remotely controlled systems. For military users, they can reduce the exposure of personnel and high-value assets to dangerous environments such as mine fields or enemy-controlled harbours and waterways. They also facilitate covert or clandestine information gathering behind enemy lines. For civilian users, the flexibility and agility of AUVs make them cost-effective sensor platforms, in particular in deeper water.
However, the actual autonomy of the vehicles in existence today is limited in many ways, restricting their potential uses. Further advances in AUV autonomy will enable new operations, such as covert, very long endurance missions (weeks) in unknown and/or hostile areas. While some experimentation is already taking place with e.g. under ice operations, the chance of failure is unacceptably high for many potential users. De-risking of long-endurance autonomous operations in unknown areas is thus an important goal for the AUV community.
The level of autonomy achieved by AUVs is chiefly determined by their performance in three areas:

Energy autonomy – reliable power sources and low power consumption for long-endurance missions.

Navigation autonomy – precise navigation and positioning with little or no position estimate error growth for extended periods of time.

Decision autonomy – the ability to sense, interpret and act upon unforeseen changes in the environment and the AUV itself.

These three areas should be addressed in a balanced fashion. In particular, navigation and decision autonomy are interlinked in various ways – the AUV's trajectory will affect navigation system and individual sensor performance, while the navigation system's performance will affect the AUV's ability to achieve the mission objectives. Actions taken by a decision autonomy subsystem can include changes to the vehicle trajectory, but also sensor configuration and utilization.

This chapter will focus on the latter two technology areas. For energy autonomy, the reader is referred to e.g. (Hasvold et al., 2006) or (Hagen et al., 2007).

2. Navigation autonomy

Precise navigation remains a substantial challenge to all underwater platforms, including AUVs. Over the last two decades, global navigation satellite systems such as GPS (Global Positioning System) have solved this issue for most surface, land and air based applications. With local or wide area augmentation systems, sub-meter positioning accuracy is available anywhere, anytime. No similar system exists for positioning below the sea surface.
Autonomous operation in deep water or covert military operations requires the AUV to handle submerged operation for long periods of time.
One philosophy is to employ the best possible inertial navigation system (INS) together with a large toolbox of aiding sensors and techniques, as done by the HUGIN navigation system (Jalving et. al., 2003).

2.1 Integrated inertial navigation system structure

An INS calculates position, velocity and attitude using high frequency data from an Inertial Measurement Unit (IMU). An IMU consists of three accelerometers measuring specific force and three gyros measuring angular rate. If left unaided the INS will, after a short period of time, have unacceptable position errors. The position error growth is determined by the class of the IMU, see Table 1. Currently the best available IMU with a feasible size for an AUV gives a position error growth in the order of 1 nmi/h (1 σ), when integrated in an INS. To reduce this error the INS needs to be aided by redundant sensor measurements. These sensors are usually integrated with the INS through a Kalman filter, which performs this integration in a mathematically optimal manner.
Fig 1 shows a schematic view of the HUGIN integrated INS, where the Kalman filter is based on an error-state model and provides a much higher total navigation performance than is obtained from the individual sensors alone.

IMU class	Gyro bias	Accelerometer bias
>10 nmi/h	1°/h	1 mg
1 nmi/h	0.005°/h	30 μg

Table 1. Classification of feasible IMUs for AUVs

2.2 DVL aided INS

The solution for most modern AUVs is a low drift Doppler Velocity Log (DVL) aided INS that can integrate various forms of position measurement updates. DVL accuracy is dependent on acoustic frequency. Higher frequency yields better accuracy at the cost of decreased range, as illustrated in Table 2. Prioritisation between range and accuracy is dependent on the application.

Frequency	Long term accuracy	Range
150 kHz	±0.5% o.s. ± 2 mm/s	425 – 500 m
300 kHz	±0.4% o.s. ± 2 mm/s	200 m
600 kHz	±0.2% o.s. ± 1 mm/s	90 m
1200 kHz	±0.2% o.s. ± 1 mm/s	30 m

Table 2. DVL range and accuracy depends on its acoustic frequency (o.s is of speed).

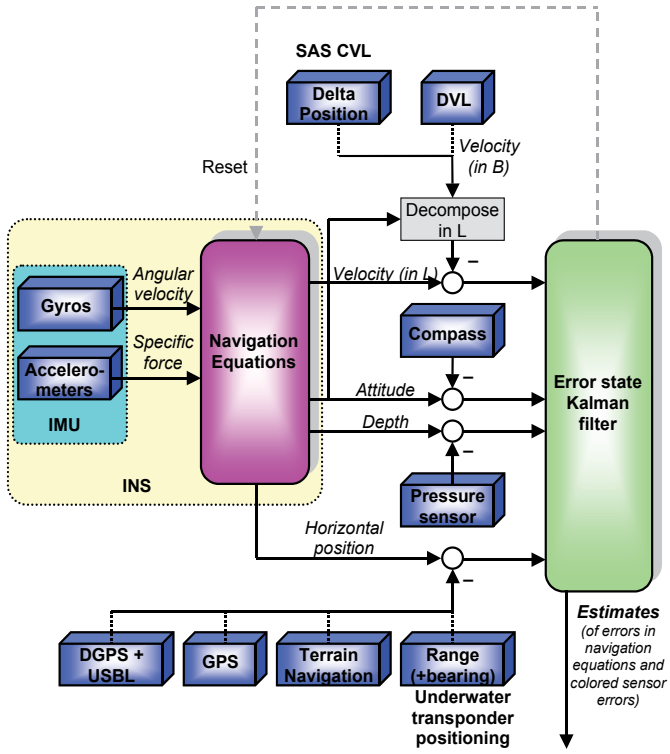


Fig. 1. Block diagram of the HUGIN integrated inertial navigation system.

The horizontal position drift in a DVL aided INS is determined by the error in the estimated Earth-fixed velocity (i.e. North and East velocity). The main contributors to this error are body-fixed velocity error, and heading error.

The error in estimated body-fixed velocity is mainly determined by the low-frequency error in the DVL itself. This error is not observable if the vehicle is travelling along a straight line and without position aiding. High frequency velocity errors on the other hand, are estimated by means of the IMU. Without aiding, even the most accurate INS will have a velocity uncertainty larger than the DVL accuracy after a short period of time.

This means that the AUV's along track position error is quickly dominated by the low-frequency error of its DVL.

Consider an AUV travelling at 2 m/s. According to Table 2, the along track error drift of a 300 kHz DVL is 28.8 m/hour, while a 1200 kHz DVL has a drift of 14.4 m/hour.

The error in heading is determined by the gyrocompassing capability of the integrated system. The heading estimation error will typically be of low frequency, corresponding to non-observable gyro bias dynamics. Referring to Table 1, a 1 nmi/h navigation class IMU typically gyrocompasses to an accuracy of $\sigma(\delta\psi) = 0.02 \text{ deg} \cdot \sec(\text{lat})$. This corresponds to an error drift of $\sigma(\delta\psi) \cdot 100$ % of traveled distance ($\sigma(\delta\psi)$ in radians). At 45° latitude, this equals 0.05% of traveled distance, or 3.4 m/hour at 2 m/s AUV speed.

The Kalman filter can compensate for part of the scale factor error of the DVL when running more complex missions than a straight line, or when position updates are available.

The expected error drift of a state-of-the-art DVL-aided INS is summarised in Table 3, both for a typical straight-line transit phase and for a typical lawnmower survey pattern.

	Position error drift (% of traveled distance)	
	Straight line	**Lawnmower pattern with 1 km lines**
Along track	0.11%	0.01%
Across track	0.03%	0.001%

Table 3. Typical position error drift for a high quality DVL-aided INS

2.3 Model and DVL water-track aided INS

As mentioned in Section 2.2, a DVL with bottom-track is an efficient aiding tool for limiting the position error drift of an INS. Even when including this capability however, situations will occur where it fails to work or measurements are discarded due to decreased quality. This will for instance occur when operating above the sensor range or over very rough bathymetry due to loss of bottom lock. In either case, in the absence of DVL bottom-track data, alternative velocity information is required to achieve an acceptable low drift INS solution between position updates. Two especially efficient alternatives are to utilize a kinetic vehicle model describing the vehicle dynamics or by including measurements from a DVL with water-track. The latter measures the velocity of the vehicle relative to the surrounding fluid. Similarly, the kinetic vehicle model predicts the same water-relative velocity by solving a set of ordinary differential equations (with measured actuation signals, e.g. propeller RPM and fin deflections as inputs). The effect of including the kinetic model in the navigation system can be seen in Fig 2, where USBL position measurements were lost for 30 minutes. The system without any velocity aiding quickly becomes useless, while the navigation accuracy in the system with model aiding remains good throughout the USBL outage.

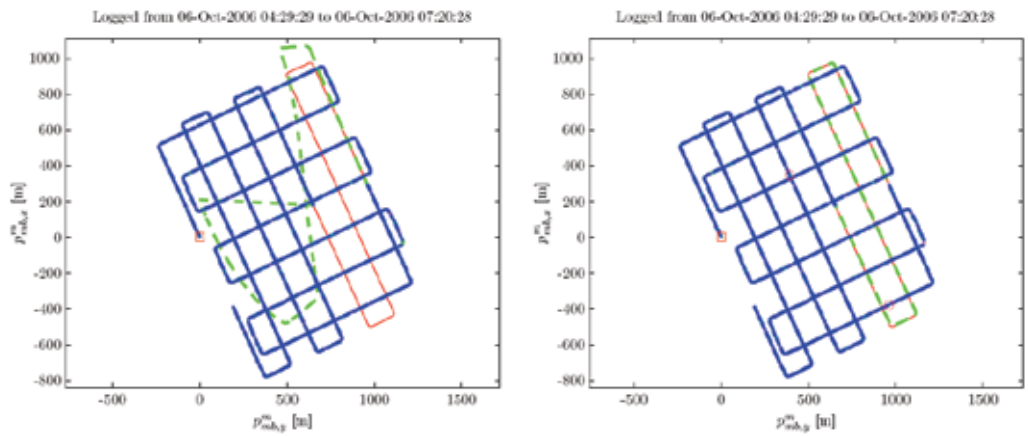


Fig. 2. Navigation during a USBL outage: The left plot shows the system without model-aiding and right when it is included. The red (solid) trajectory is considered ground truth. The blue (o) data show the USBL measurements. The segment without measurements corresponds to 30 minutes. The green (dashed) is the real-time navigation solution.

Note that when aiding the INS with both DVL water-track measurements and data from the kinetic model, it is important to also include real-time sea current estimation in the

navigation system. The sea current will often constitute the dominant error source when aiding the INS with water-relative velocity data, in particular when applying DVL water track data. For additional information on model-aiding and sea current estimation, the reader may refer to (Hegrenæs et al., 2007) and (Hegrenæs et al., 2008).

2.4 Micro delta-position aiding

Some of the sensors carried by an AUV can observe the same part of the seafloor in consecutive sensor measurements, such as a Synthetic Aperture Sonar (SAS), a 3D sonar, or a camera. By correlating these consecutive measurements of the same seafloor patch over a small time interval, a measurement of the change of the AUVs position between these measurement times may be derived. A Correlation Velocity Log (CVL) is another sensor that provides a similar measurement, although using a different measurement principle.
These sensors provide a different measurement than velocity to integrate with the INS. This aiding technique is herein called micro delta-position aiding.
When the time interval is very short, the measurement more and more resembles a pure velocity measurement. This technique therefore provides redundancy to the DVL, but may also decrease the position error drift of the DVL-aided INS substantially if the measurement is sufficiently accurate. In this aspect, SAS is a particularly attractive sensor. SAS uses consecutive pings to synthesise a larger array. For this to work properly, the array position displacement between pings must be found with extremely high accuracy. This is called *micronavigation* or DPCA (displaced phase centre antenna) (Belletini & Pinto, 2002; Hansen et. al., 2003). By using this position displacement as a velocity measurement, it can potentially be an order of magnitude better than even the most accurate DVL.
The technique is fully autonomous, but requires a suitable sensor onboard the AUV.

2.5 GPS surface fix

As seen in Fig. 1, several alternatives may be included for providing the integrated inertial navigation systems with position updates. Obtaining GPS surface fixes is the most intuitive method and should be used whenever practical. The time between surfacing depends on the required navigation accuracy and additional aiding sensors. The following GPS positioning services can be used:

- GPS Standard Positioning Service (SPS)
- GPS Precise Positioning Service (PPS)
- Differential GPS (DGPS)
- Real-Time Kinematic GPS (RTKGPS)

GPS SPS is available to all users worldwide. GPS PPS is available only to authorised users and primarily intended for military purposes. GPS PPS receivers should be the choice for military AUVs, at least for operations in denied areas. Compared to SPS, PPS is more resistant to jamming and deception. SPS and PPS have comparable accuracy. AUVs for detailed seabed mapping will typically be equipped with DGPS, or in some cases even RTKGPS. An example of a mission where GPS was applied is shown in Fig. 3.

2.6 Combined GPS-USBL (Ultra Short Base Line)

In deep water seabed mapping, deploying and following the AUV with a survey vessel is the preferred method for obtaining maximum position accuracy. The survey vessel is

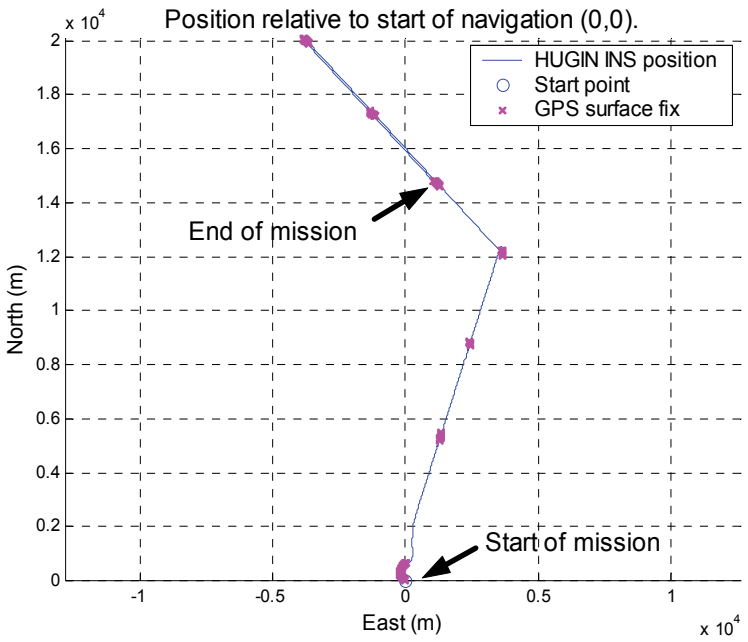


Fig. 3. Mission example with GPS surface fix using a HUGIN AUV, September 2003.

equipped with a GPS system, and tracks the AUV with an ultra short base line (USBL) positioning system, e.g. Kongsberg HiPAP. By combining GPS with USBL, a global position estimate can be obtained, which is then transmitted to the AUV using an acoustic link. The accuracy of the position estimate depends on both the GPS and USBL precision. The position accuracy of single measurements from the standalone HiPAP is in the order of 0.2% of the range (line of sight from the surface ship to the AUV).

While this positioning source will not be available in autonomous operations, it is very useful for providing accurate initialisation of the navigation system when operating in deep water. Transmitting a series of GPS-USBL position fixes to the AUV will yield a very low initial position estimate error (typically on the order of 1 m), and a very low heading estimate error – which in turn minimizes drift when no position aiding is available.

The navigation precision obtained by combining DGPS-USBL and DVL with INS was demonstrated with the HUGIN 3000 AUV in an accuracy test carried out by C&C Technologies in the Gulf of Mexico in October 2000. The test was carried out at 1,300 m depth, and eleven different headings were used (five of the lines were mapped in opposite directions) when mapping a wellhead to maximize the visibility of any correlated errors following the AUV or ship. When post-processing the navigation data and wellhead observations in NavLab (see Section 2.11), a position accuracy of 1.2 m and 1.7 m (1σ) north and east was obtained. The final map showing the wagon-wheel like trajectory and the experimentally validated navigation precision (the exact location of the wellhead was known in advance) are shown in Fig. 4. An absolute position accuracy of 4 m (1σ) was later obtained during successive navigation accuracy tests carried out at 2,100 m depth.

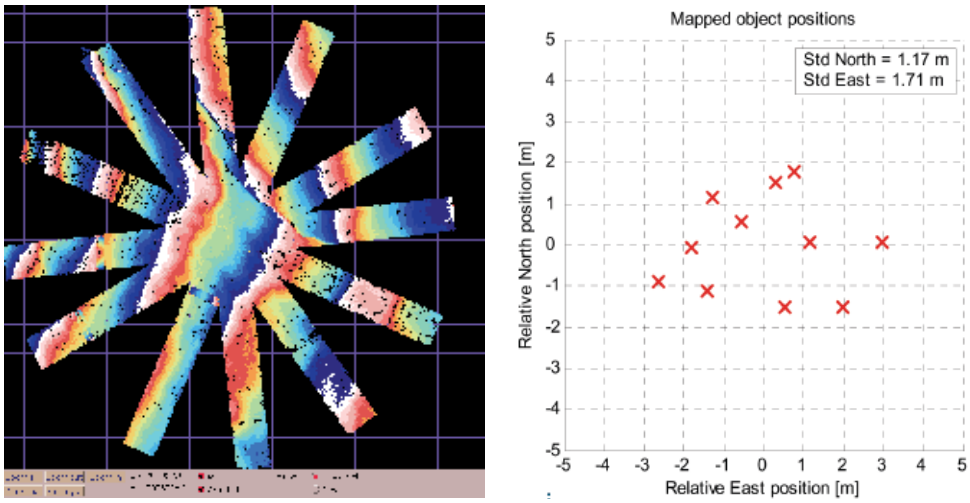


Fig. 4. A wellhead mapped repeatedly with different headings to evaluate the position accuracy of the final map, and hence the navigation system. The true position of the wellhead was located at coordinates (0,0) relative north and east.

2.7 LBL (Long Base Line)

Underwater transponders may be used to provide submerged position measurements to the AUV. If multiple transponders are arranged in a network for this purpose, it is called a long base line (LBL) system. The AUV interrogates the LBL network, and uses the reply from each transponder to calculate the range from the AUV to the transponders. If the 3D position of each transponder is known, the AUV can compute a unique position by triangulation if replies are received from three or more transponders. If only two transponders are within range (sparse LBL system), two possible position solutions exist. These are mirrored about the base line between the transponders.

The LBL network is deployed and calibrated by a surface vessel, and is therefore not a strictly autonomous solution. However, once the network is established, the AUV may navigate autonomously with bounded error drift by visiting the network occasionally.

2.8 UTP (Underwater Transponder Positioning)

Range measurements from underwater transponders may also be coupled tighter with the AUV's aided INS. This concept, called underwater transponder positioning (UTP) or synthetic baseline, has the advantage that it will work with even a single transponder. An example illustrating the deployment of a transponder is shown in Fig. 5. The transponder location can be determined within 1 m relative global datum (e.g. WGS-84) at 1000 m depth if applying a Kongsberg HiPAP USBL system during the box in process.

As the AUV moves around or between transponders, new ranges from different directions will be obtained. Combined with INS, this can provide a navigation accuracy comparable to using USBL aiding from the support vessel. Fig. 6 shows the trajectory of a HUGIN AUV and UTP transponder locations. HUGIN navigated with UTP as the only source for position updates. A 30 kHz UTP system with approximately 1.4 km range was used. The AUV ran two straight lines of 7.5 km with two transponders 4 km apart. With NavLab navigation post-processing, the navigation data was compared to independent USBL data stored on the

survey vessel. The difference between the UTP post-processed navigation solution and the independent USBL post-processed navigation solution is shown in Fig. 7. Since the deployment of transponders is time consuming, an important feature of UTP aided navigation is that only one transponder is required to bind the INS position error drift. While the system works with only one transponder, any number of transponders may be incorporated in an optimal way. Compared to a traditional LBL system (Section 2.7), UTP has improved accuracy due to tight coupling with the INS, increased operating area and significantly lower deployment costs. Further discussion on the use of UTP and risk reduction in autonomous under ice AUV surveys can be found in (Jalving et al., 2008).

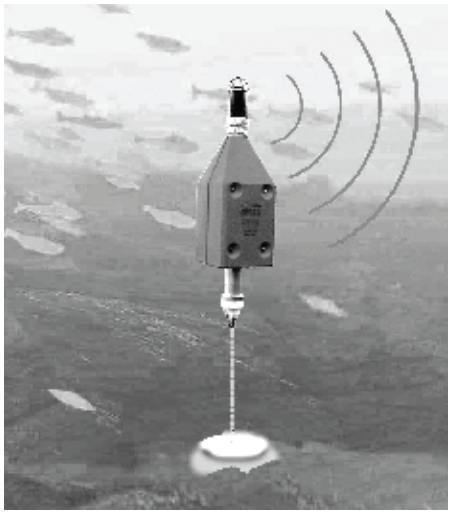


Fig. 5. A Kongsberg Maritime release transponder

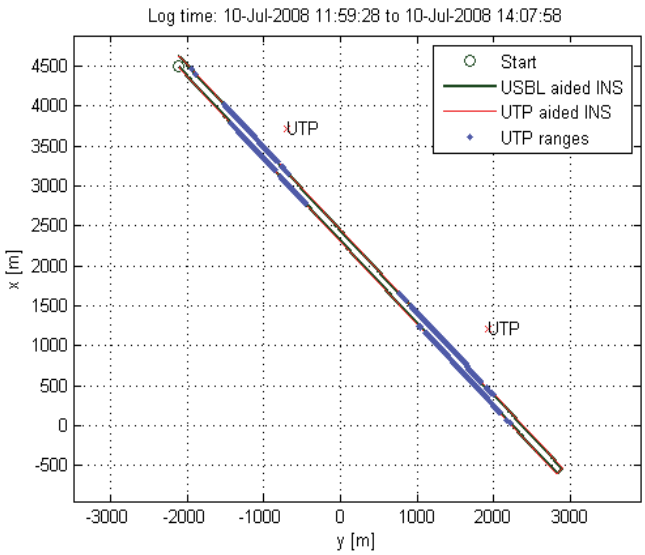


Fig. 6. North (x) and east (y) relative coordinates of 2D AUV trajectory during sea trial. Two UTP transponders were deployed 4 km apart.

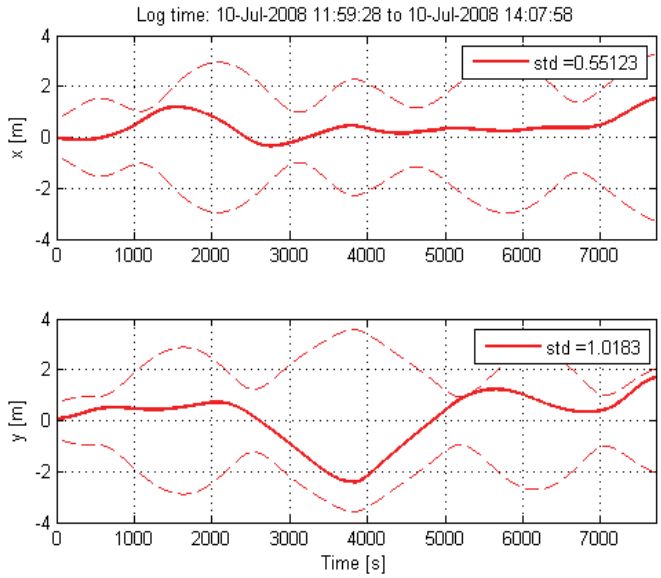


Fig. 7. HUGIN position accuracy with UTP and DVL aiding. Solid red: difference between UTP post-processed navigation solution and independent HiPAP USBL post-processed navigation solution. Red dashed: predicted 1σ position uncertainties in navigation system with UTP.

When aiding the AUV with range measurements, the vehicle only needs one transducer. By using two vehicle mounted transducers, it is also possibly to obtain bearing measurements. The baseline (distance) between them should be as long possible (in principle this becomes a short baseline system, SBL). Similar to the range measurements, the bearing measurements are also tightly integrated with the INS, enhancing the overall accuracy even further.
A possible development is to have the AUV deploy the transponders itself. Once deployed, the current generation of Kongsberg transponders have battery capacity for two to three years operation; upcoming transponders will have a battery capacity of five years. This way, the AUV will be able to deploy a trail of underwater acoustic buoys for UTP navigation. These transponders can also be used as acoustic communication relays. This concept can be denoted concurrent deployment and navigation (CDN).

2.9 Terrain navigation

Some of the typical sensors carried by an AUV provide bathymetric measurements; either as a main product or by-product. Among these sensors are DVLs, multi beam echosounders, altimeters, and interferometric side scan sonars. The bathymetric measurements from these sensors can be correlated with a pre-obtained digital terrain model (DTM) of the seafloor, and as a result the AUV position within this DTM may be estimated. This technique is called terrain navigation. The algorithms performing the correlation can conceptually be divided into global search algorithms and tightly integrated algorithms.
Examples of global search algorithms are the Terrain Contour Matching (TERCOM) (Golden, 1980), the Point Mass Filter (PMF), and particle filters (Bergman et. al, 1999). There are different degrees of sophistication to each of these algorithms, but in essence they take

input from the INS, a bathymetric sensor and a DTM, and estimate a global position measurement to be integrated back with the INS. These algorithms handle highly non-linear bathymetry with great results, but have convergence problems in terrain with little variation.
The tightly integrated algorithms, such as Terrain Referenced Integrated Navigation (TRIN) (Hagen & Hagen, 2000), and Sandia Integrated Terrain Aided Navigation (SITAN) (Hostetler, 1978), integrate the bathymetric range measurements tightly with the INS. These algorithms handle linear and weakly non-linear terrain very well, but may have some problems with highly non-linear terrain.
Terrain navigation is a fully autonomous technique, but it requires a DTM of the mission area. This information is unfortunately not always available. A solution to this problem is called Concurrent Mapping and Navigation (CMN), where the DTM is made by the AUV in-situ, and used concurrently by the terrain navigation algorithms to bind the position error drift.

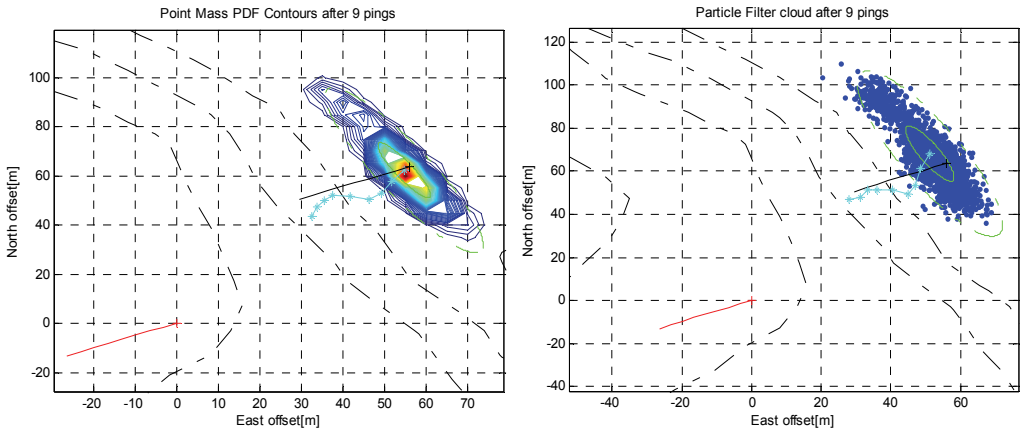


Fig. 8. The contour lines of the PMF's probability density function (left) and the particle filter cloud (right) after processing 9 pings from a DVL with the tool TerrLab (Hagen, 2006). Red line is a priori navigation solution, black line is true position and cyan marks position estimates from the terrain navigation algorithms.

2.10 Macro delta-position aiding

As an extension of the idea of the micro delta-position aiding, we can consider sensor measurements where the same patch of the seafloor is seen with an arbitrarily large time interval between the measurements. As an example, consider the detection of mine-like objects from sidescan images. If the AUV passes the same object multiple times, and the localisation accuracy of the object is mainly contributed to the AUVs navigation accuracy, then the difference between the object's estimated position from the two observation times gives a valuable measurement of relative position error of the AUV. This type of measurement can be integrated with an INS, and the technique is herein called macro delta-position aiding. Another example, closely related to CMN, is to use sparse bathymetric information from an interferometric sonar covering overlapping regions during the mission. After running one of the terrain navigation algorithms, we will be able to make a macro delta-position measurement for the INS.

This technique is fully autonomous, but requires an imaging or bathymetric sensor. It is also only possible to bind the position error drift with this technique. The initial global error of the AUV is not observed, as can be done with regular terrain navigation

2.11 Navigation post-processing

When discussing navigation in autonomous operations, it is important to distinguish between performance in real time and after post-processing. The real-time performance obviously determines where the AUV actually collects its data. Depending on the application, (e.g. when creating bathymetric maps or during mine reconnaissance) it may be desirable to enhance the navigation precision further in post-processing. This is standard procedure for the HUGIN AUVs, where the post-processing is carried out using NavLab (Gade, 2004).

Post-processing will improve the estimation accuracy as well as the integrity. The increased accuracy is due to the use of smoothing, which is an optimal estimation technique that utilizes both past and future measurements. Smoothing is especially effective when position updates are scarce, which is the case with GPS surface fixes, terrain navigation with few reference areas, and scattered underwater transponders. The effect of navigation post-processing when running a 15 km straight-line trajectory with GPS fix at the end is shown in Fig. 9. The effect is smaller, but still significant, when running a lawnmower pattern. Another important feature of navigation post-processing is increased navigational integrity and increased ability to recover faulty data sets. For one thing, wild point filtering can be more reliable when both past and future measurements can be considered. The smoothing algorithm is in general more robust against degraded sensor performance, hence efficiently improving the final navigation accuracy and reliability in such cases.

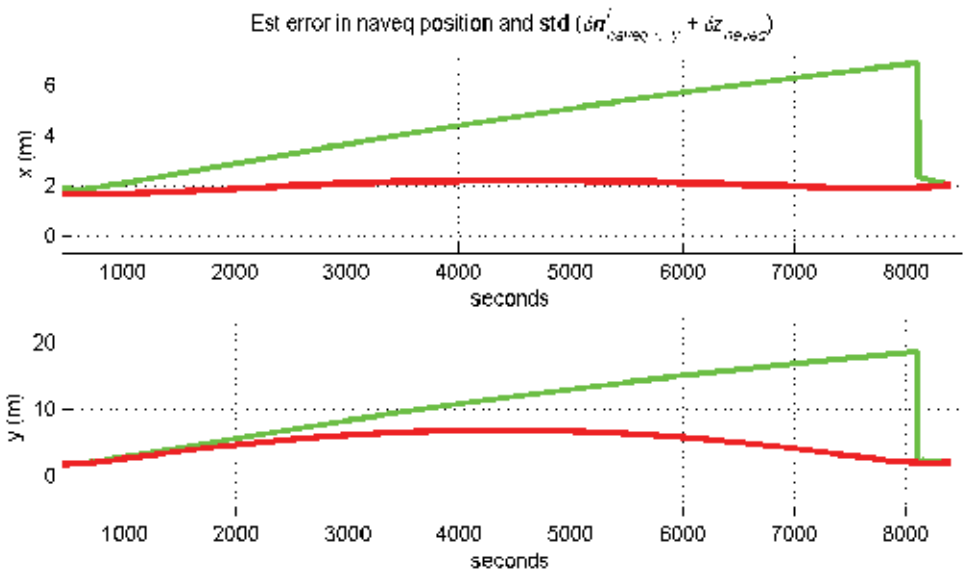


Fig. 9.The effect of navigation post-processing when running a straight trajectory with GPS fix every 15 km. Green graph: predicted real-time position accuracy (1σ). Red graph: predicted post-processed position accuracy (1σ). The true navigation errors have been experimentally validated to be consistent and within their predicted uncertainties.

3. Decision autonomy

The vast majority of today's AUVs operate with a pre-programmed mission plan specifying waypoints and vehicle parameters for the entire mission. Complex tasks that cannot be accurately specified in advance must be solved through intermittent communication with a human operator. This obviously limits the performance and applicability of such vehicles. A truly autonomous vehicle must be able to perceive its own condition and its environment, and respond appropriately to unexpected or dynamic situations. Updated situational awareness requires an extensive set of sensors and data analysis tools, but the most challenging part of decision autonomy is still to select advantageous actions based on the information available.

Conceptually, decision autonomy can be divided into two categories: The ability to handle internal malfunctions (sustainability) – and the ability to handle unpredictable external events (adaptivity). The latter is important for optimizing mission execution by adaptive, real-time mission planning based on e.g. observed bathymetry and sea current conditions. It also facilitates novel applications such as adaptive data collection and cooperation with other vehicles. Sustainability is vital to realise both long-endurance missions, as the probability of sub-system failures increases with mission duration, and missions in extreme environments, e.g. under ice, where consequences of failures may be unacceptable. In all cases, actions must be chosen so that the overall mission goals are achieved to highest degree possible. These actions may include modifying the mission plan, and algorithms for automated re-planning are thus required.

Section 3.1 provides an overview of the different architectures that are common to use when designing an autonomous control system. Section 3.2 introduces autonomous path planning, explaining the consequences of the control architecture on the complexity of the planning algorithms. Planning requires knowledge about the environment of the vehicle, and Section 3.3 discusses ways to represent this knowledge. For the vehicle to be truly autonomous, it is important that its sensors are also able to operate autonomously; this is discussed in Section 3.4. The ability to handle internal malfunctions is covered in Section 3.5. The autonomous vehicle may be required to cooperate with other vehicles, both autonomous and not. Some scenarios requiring such cooperation are given in Section 3.6. Finally, an example of an autonomous vehicle is given in Section 3.7, which very briefly describes the autonomy system of the HUGIN AUV.

3.1 Control architectures

A control architecture is a software framework that manages both sensor and actuator systems, thus enabling the AUV to perform a user-specified mission. There are basically three different approaches to control architectures: deliberate, reactive and hybrid systems (Arkin, 1998)(Valavanis et al, 1998)(Ridao et al., 2000)(Russell & Norvig, 2003).

Deliberate (hierarchical) architectures are based on planning using a world model. They allow reasoning and making predictions about the environment. Data flows from sensors to the world model, which is used together with a predefined set of mission goals to plan new actions to be undertaken by the actuators (Fig 10). This "Sense-Plan-Act" scheme is suited for structured and predictable environments, and its well-defined, tightly-coupled structure simplifies the process of system design and verification. However, problems arise when trying to maintain an accurate, real-time world model in the complex, dynamic and only partly observable environments typical for underwater operation. System response may then be too slow or erratic.

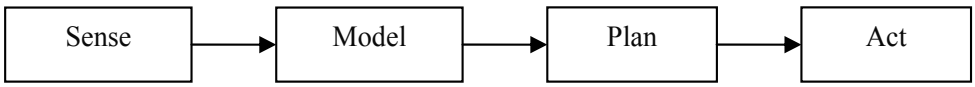


Fig. 10. The deliberate control architecture is based on world modelling and planning

Reactive (also called behavioural) control architectures are based on the simple "Sense-React" scheme and typically involve neither world model nor planning. Basically, a group of parallel behaviours act independently generating outputs to the actuators according to their input sensor data (Fig. 11). A mission is normally described as a sequence of phases each with a set of active behaviours. A "Transit" phase could for instance have the active behaviours "Avoid collision", "Go to waypoint" and "Error handling". The system's global behaviour emerges from the coordination of the elemental active behaviours and its interplay with the environment.

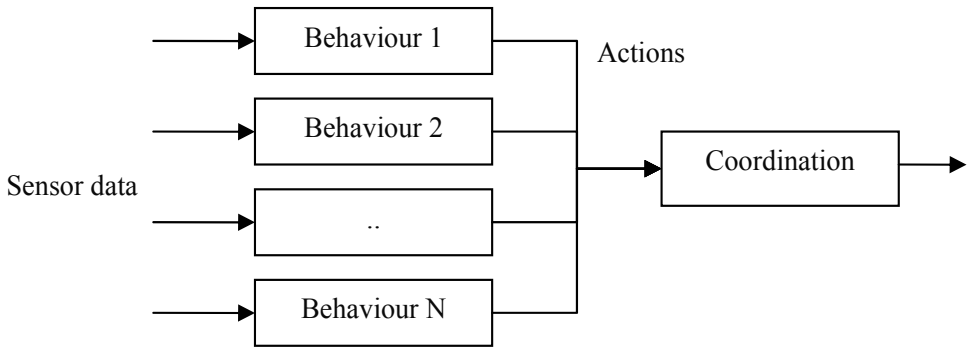


Fig. 11. The reactive control architecture is based on independent, parallel behaviours

Since each behaviour pursues its own goal, behaviours may issue contradicting actions, making behaviour coordination an important part of the reactive system. The different methods of behaviour coordination can be divided into a cooperative and a competitive approach (Carreras et al., 2000). In the cooperative approach, the behaviours vote on desired actions (e.g. the desired direction to move). The resulting action can then be either the one with the most votes, or a weighted average of the behaviour outputs. The votes can be multiplied by a gain value according to the priority of the voting behaviour. It is also possible to implement negative votes, or even veto votes, to avoid making the vehicle move into forbidden or dangerous regions.

The competitive approach to behaviour coordination has a single behaviour in control of the vehicle at any given time. An example is the well-known subsumption architecture (Brooks, 1986), where higher-prioritized behaviours will override (subsume) lower-priority behaviours when activated. For instance, an "Avoid collision" behaviour will subsume a "Go to waypoint" behaviour when the vehicle is approaching an obstacle.

Competitive behavioural coordination provides more robustness than cooperative coordination since safety behaviours may act undisturbed in dangerous situations. As only one behaviour acts at a time, the competitive approach is also more modular and easier to tune. Cooperative behaviour coordination, on the other hand, tends to give more optimized paths. This can be seen by considering that the "Avoid collision" behaviour will be able to

manipulate the vehicle path at an earlier time, making the resulting path smoother and less shortsighted. The cooperative approach also allows for working on several goals simultaneously, which is not possible with the competitive approach.
Reactive control is sensor-driven and appropriate for making low-level decisions in real-time. However, reactive control rarely yields a plausible solution at the global level, because global control decisions depend on information that cannot be sensed at the time of decision making. For such problems, deliberate control is more appropriate. Furthermore, it can be hard to predict the overall vehicle behaviour when reactive control is used. Consequently, most AUV architectures use reactive techniques at the lower levels of control with deliberate techniques at the higher levels. Such architectures are called hybrid architectures. The most popular is the three-layer architecture, which consists of a reactive layer, an executive layer and a deliberate layer (Gat, 1998)(Brutzman et al., 1998) as seen in Fig. 12.

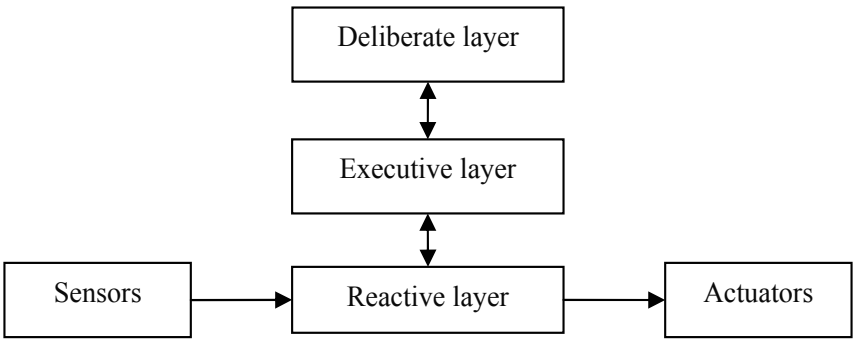


Fig. 12. The hybrid control architecture combines reactive and deliberate control.

The executive layer serves as a supervisor, accepting directions from the deliberate layer and sequencing them for the reactive layer. Similarly, it receives status and processed sensor data from the reactive layer and sends invocations to the deliberate layer. Decision cycle is usually on the order of milliseconds, one second and minutes for the reactive, executive and deliberate layer, respectively (Russell & Norvig, 2003).

3.2 Path planning

Regardless of the control architecture chosen, some sort of planning is typically necessary to maintain safety and efficiency of the system. Planning can be done both before and during the mission (replanning). By replanning the path, the vehicle is able to account for changes in its internal status and in the environment. The plan typically constitutes an explicit path to follow, but can alternatively be in the form of an artificial potential field used by the vehicle to guide it (Tan et al., 2004). At the very least, the plan may consist of a sequence of behaviour parameter sets for a reactive controller. Replanning will then be done by changing the path, adding or removing repulsors or attractors from the artificial potential field or changing the parameter set sequence. Regardless of the implementation, it is important that the planning mechanism is able to make the vehicle avoid hazardous areas, as well as to achieve the goals of the mission.
The required level of path plan detail is dependent on the control architecture of the vehicle. A purely deliberate architecture needs a fully detailed plan, while a reactive architecture

requires no other plan than perhaps a sequence of behaviour parameters. A hybrid architecture will need a plan only for the top, deliberate layer, while the lower reactive layer will take care of small adjustments and anti-collision. Thus, the hybrid architecture yields a simpler planning task than the deliberate architecture.

Planning premises are also set by the performance of the navigation system. If the navigation uncertainty is large, a robust plan that has good clearance from obstacles is necessary. Furthermore, the fine detail of a plan becomes useless if the vehicle does not have sufficient navigation capabilities to follow it. Thus, navigation autonomy is a critical feature for autonomous systems that rely on planning. If a plan can be specified relative to the vehicle's current position, local (relative) navigation accuracy is the important parameter. This is advantageous, as local navigation is more accurate than global.

There is a variety of methods available to generate explicit paths (Lavalle, 2006). One method is to convert the free area of the environment into a graph, as shown in Fig. 13. A graph search algorithm can then be applied to find a favourable path based on some optimization criteria. An example is Dijkstra's algorithm (Dijkstra, 1959) which finds the shortest path between two points. Robustness to navigation uncertainty can be introduced by reducing the size of the free areas through safety margins. If the environment is not known in advance, then a dynamic graph search algorithm can be employed (such as the D* algorithm (Lavalle, 2006)).

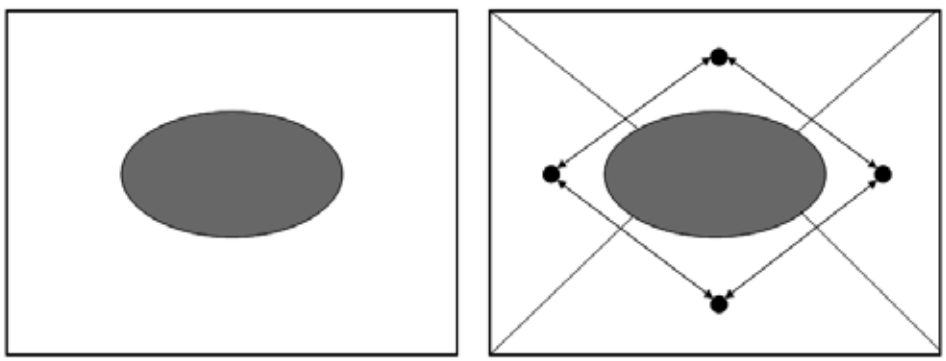


Fig. 13. Converting an area with an ellipsoid obstacle into a grapth with 4 vertices.

A path may be required to be feasible, i.e. it must adhere to the dynamic constraints of the vehicle. Such constraints must then be included in the graph search algorithm. One way to do this is to transform the environment into a multidimensional space, where each point is a possible configuration (state) of the vehicle. This space is known as the configuration space, or state space (Lavalle, 2006). If this space is discretized, a graph searching algorithm is straightforward to implement. Creating a configuration space, and transforming obstacles into this space, is not a trivial task and may require a significant amount of computational resources. Furthermore, for complex vehicles the configuration space has many dimensions (one for each degree of freedom), increasing the amount of time needed to perform a successful search. Due to the complexity of the configuration space, dynamic objects are also hard to implement.

It is also possible to use dynamic programming to solve the path finding problem. Methods from optimization theory (Nocedal & Wright, 1999) can then be applied to find a plan that is

optimized with respect to some objective function (e.g. by minimizing the amount of energy used). The objective function may include constraints, and the optimization methods are not dependent upon a discretization of the environment. There are, however, several drawbacks with using optimization algorithms as well. The optimization problem is not trivial to formulate, and the algorithms can use much time and computational resources in computing a solution, which still may not be globally optimal. Further, most algorithms require a feasible starting path, which again might be difficult to compute.

For underwater vehicles, the necessity of a strictly feasible path should be carefully considered. The environment is typically a large 3D volume, sparsely populated with obstacles and rarely adequately known in advance. In this case safety margins around the path will keep the vehicle safe, even if the vehicle is forced to move outside of the path due to dynamic constraints. This way of avoiding dynamic constraints on the path considerably simplifies path planning.

The planning complexity can also be reduced through implicit path planning which does not create an explicit path for the vehicle to follow, but rather transforms the environment into an artificial potential field (Tan et al., 2004). In this field, obstacles and other forbidden areas serve as repulsors, pushing the vehicle away from them, while goals serves as attractors pulling the vehicle towards them. The vector sum of influences on the vehicle acts as input to the vehicle's low level control system. This method is robust for errors in global navigation, and is quick to compute. However, like all planning algorithms, it is dependent on either a completely known environment or on sensors capable of detecting obstacles. Further, the vehicle can become trapped in local minima if care is not taken to remove this (or detect entrapment). Fig. 14 shows an artificial potential field where the goal is located next to a wall.

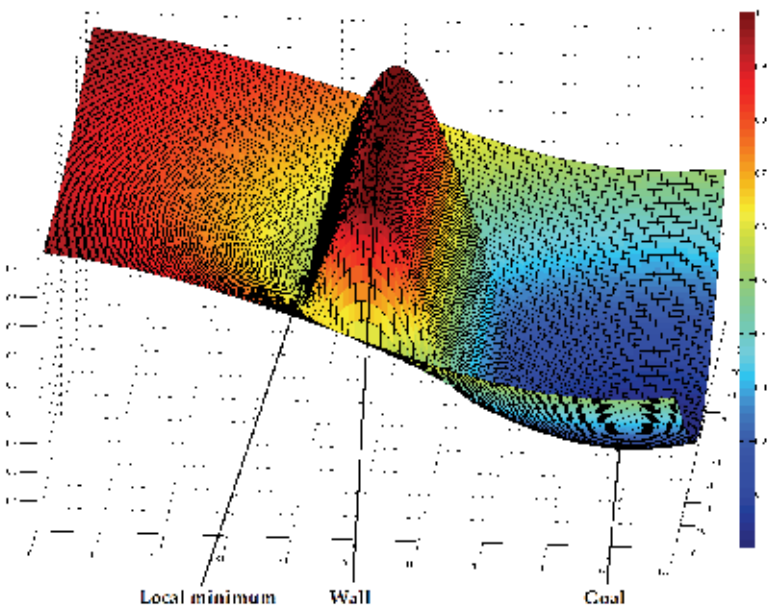


Fig. 14. An artificial potential field with a goal position beside a wall. Note the local minimum on the opposite side of the wall.

For mission goals other than just transiting to a position, path planning can become considerably more complex. An example is area coverage, in which an area is to be mapped or imaged with a sensor (e.g. a sonar, camera or echo-sounder). If a graph search algorithm is used, the problem resembles the travelling salesperson problem (Bonabeau et al., 1999). Dynamic programming and artificial potential fields also become rather complex for more advanced tasks. In this case, it can be advantageous to make use of assumptions and heuristics in the path planning. For coverage planning, which is an area of much research interest (Bourgeois et al., 1999)(Hert et al., 1996)(Huang, 2001)(Choset, 2001), the simple lawnmower pattern is typically suitable. The coverage planning problem for an obstacle-free environment is then reduced to finding the line spacing, orientation and altitude of the pattern. Obstacles can be dealt with by dividing the environment into obstacle-free cells (Huang, 2001).
If the path-finding algorithm is not too computationally expensive, it is possible and advantageous to replan the path during the mission. The ability to replan the path makes the vehicle able to account for internal and external changes, and removes the demand for an a priori known environment. For example, for coverage planning using a lawnmower pattern, the pattern can be replanned in each turn to optimize track spacing based on actual seafloor coverage, or search direction based on discovered bathymetry and sea currents (Hagen et al, 2007). The plan must also be replanned to account for inputs from other parts of the system, such as when the navigation system requests a GPS fix to be done.
Some missions cannot, by definition, have an explicit, planned path for the vehicle. An example is adaptive data collection, where the vehicle may perform additional measurements on objects found in the sensor data or may follow a sensor data gradient (e.g. temperature, salinity or plankton density). In both cases, the resulting path is completely dependent on the sensed environment. Such missions must be goal-oriented. If planning is employed, replanning is necessary each time the gradient estimate is updated or an object is found. The frequency of replanning may place severe limitations on the computational time allowed for the planning algorithm, especially in the case of gradient following. If the limitations are too strict, the planning task must be reduced by giving more responsibility to the reactive layer.

3.3 Knowledge representation

Non-trivial path planning requires some input knowledge about the environment. Both a priori knowledge, e.g. about obstacles and bathymetry, and data gathered during the mission can be useful. This geographical information can be presented as map layers and typically includes parameters like (Midtgaard et al., 2006):

- Water depth (a priori; updated during the mission)
- Ice cover depth
- Obstacle locations (and possible classifications)
- Objects of interest (e.g. mines or wrecks)
- Forbidden/dangerous zones
- Reliability/age/accuracy of information
- Sea currents
- Seabottom type

Traditionally, significant amount of work has been spent on developing memory-efficient representations of maps, such as the quadtree, TIN models and spatial decomposition (Tan

et al., 2004). While these methods may reduce memory requirements by several orders of magnitude without loss of information, the time to access information can increase dramatically. Current start-of-the-art computer systems have enough memory storage to make a uniform grid representation a reasonable choice.
For a deliberate architecture, dependent on a high level of planning, the required grid spacing of the map is set in a large part by the environment. If the environment is simple with a sparse obstacle density, only a coarse map is required. However, a complex environment (e.g. a harbour) requires a fine-gridded map. A reactive architecture may not need a map at all, or only a very coarse one, regardless of the environment. A hybrid architecture, with a top deliberate layer and a bottom reactive layer, only needs a plan for the deliberate layer. Thus, it only requires a coarse gridding on the map in any environment. How coarse is dependent both on the mission an on how much of the control is performed by the deliberate and how much is performed by the reactive layer.
Note that the product of the mission, e.g. a bathymetric map, will typically be stored in another, high-resolution format that need not be kept in memory.

3.4 Sensor autonomy

To the end user, an AUV is nothing more than a platform carrying the sensors (or other payloads) that perform the actual objective of the mission; data collection. In fully autonomous HUGIN missions, the performance of the onboard sensors and other subsystems are normally checked during the initial phase of the mission, where the AUV is within communication range of the surface vessel. Using either acoustic or RF links, the operator can ensure that all subsystems perform as assumed during the planning phase – and take corrective action as needed. If conditions are known to vary, different sensor settings may be pre-programmed for different parts of the mission plan.
However, when operating autonomously in areas where the environmental conditions are unknown or rapidly varying, this may not be sufficient and the AUV will need to re-adjust the sensor configuration autonomously.
Some sensors include some functionality for automatic adaptation to varying conditions; some are very simple and have no parameters that can be changed. However, most sensors available today are designed based on the assumption that a human operator will monitor the sensor data and adjust the sensor parameters as needed. This is in particular true for more advanced sensors such as side scan sonars, multibeam echo sounders, synthetic aperture sonars and optical imaging systems.
Over the following paragraphs, we will use the Kongsberg HISAS 1030 interferometric synthetic aperture sonar (SAS) as an example (Fossum et al., 2008). This sensor is used for very high resolution imaging (2-5 cm both along- and across-track) and high resolution bathymetry out to approximately 10 times AUV altitude.
HISAS 1030 transmits wide-beam, wide-band pulses to port and starboard side at regular intervals; typically, a few Hz. Signals reflected back from the seafloor and from objects in the water volume are received in two long multi-element receiver arrays, as well as in the transmitter. A complex signal processing chain, working on data from a number of consecutive pings, transforms the raw data into sonar imagery and bathymetry (Hagen et al., 2001)(Hansen et al., 2003).

The performance of a SAS is limited by a wide range of factors – AUV altitude, stability of motion, navigation accuracy, self noise, and sound speed variations, to name a few.
In shallow water, performance is often limited by acoustic multipath; i.e., simultaneous reception of signals that have travelled different paths from transmitter to receiver (reflected from the seafloor, from the surface, seafloor then surface, surface then seafloor, etc). A fully autonomous SAS would analyse the returned signal and, based on the sonar data and knowledge of the environment, tune operation to optimize performance. The most obvious candidate parameters to modify include the transmission beam width and direction, and the AUV altitude. After applying these changes, the autonomy system may determine that performance is still inadequate; e.g., the actual range to which good sonar imagery can be produced is less than required to achieve full bottom coverage. This will then trigger replanning of the mission, by spacing survey lines tighter.

3.5 Error handling

Robust error handling is vital to achieve sustainability and survivability of the vehicle. A conservative approach is to make the vehicle perform an emergency ascent when encountering any severe error or abnormal situation. This will ensure that the vehicle can be recovered, even though the result is mission failure.
For under-ice or covert missions, an emergency ascent procedure will not be possible. Ensuring survival of the vehicle in emergency situations thus becomes significantly harder. In covert or clandestine operations, preventing discovery may be much more important than recovering the vehicle. In many scenarios, one option may be to perform an emergency descent, allowing the vehicle to rest at the sea floor until a recovery operation may be possible.
Clearly, a more intelligent error handling will be needed for long endurance and complex missions. This will increase survival capabilities in dangerous situations as described above, and also enable the vehicle to complete the mission to the best degree possible. A challenge with error handling algorithms is that the error may be in the algorithm itself. This makes redundancy important, and necessitates a bottom-line error handling procedure with emergency ascent/descent.
Intelligent error handling requires the vehicle to perceive both its external environment and internal status. By measuring sensor performance it is possible to alter the mission plan to optimize performance, as described under path planning. The vehicle should also be able to estimate the remaining time of operation before it runs out of energy, ensuring that it always has enough energy for the return journey.
The error handling can be divided in three layers. The topmost layer is concerned with achieving the mission goals as best it can under the circumstances. This layer will be mission-dependent, and a different software module may be needed for each kind of mission. The middle layer is mission-independent and is concerned with getting the vehicle to the specified rendezvous position. The mission will not be a success, but the vehicle will at least be recovered at a safe location. The bottom layer is the last line of defence, and will attempt to ensure survivability of the vehicle. It is this layer that will perform emergency ascent/descent. The importance of the bottom layer makes it advantageous if this is implemented both in software and in hardware (e.g., if the power system suddenly fails, the hardware solution will ensure that an ascent procedure is initiated).

3.6 Cooperating systems

Cooperating systems is a rising topic in the field of underwater intelligent vehicles providing many potential benefits. The systems can all be AUVs, or be a mixture of underwater and surface vehicles. Some vehicles may be autonomous while other may be remotely controlled or manned. Below are some examples of cooperation.

The mission time to survey a given area may be significantly reduced by using several autonomous vehicles. The level of communication between these vehicles depends on the level of planning involved (and thus on the control architecture). If each vehicle is given a part of the area to explore independently, no intervehicle communication is necessary. However, it is also possible for the vehicles to interact, for instance by transmitting their positions and sensor findings as guidelines for the other vehicles. This is of particular interest for adaptive data collection, where the combined efforts of all the vehicles can be focused through either directions from a master AUV or control algorithms inspired by swarm intelligence (Bonabeau et al., 1999). The use of several AUVs moving in formation may be advantageous when tracking sensor data gradients, as calculating the spatial gradient based on data from a single vehicle requires intensive manoeuvring.

Two AUVs with different sensor suites can cooperate in a mission where the objective is to detect and examine certain objects or features on the seafloor. The first vehicle is equipped with a sensor for long-range detection and covers the complete area using an efficient survey pattern (e.g. lawnmower). Sensor data is automatically processed in the vehicle, and results transmitted over acoustic link. The second vehicle is equipped with close-range inspection sensors and visits only the detection locations transmitted by the first vehicle.

An AUV can cooperate with an accompanying surface ship in several ways. The surface ship (manned or unmanned) may provide an accurate position update to the AUV, using GPS and USBL as described in Section 2.6. The AUV can also acoustically transfer key parameters and data subsets to an unmanned surface vessel, which can use wireless communication to receivers located on land or in air, thus substantially increasing the effective communication range for the AUV. Another mode of cooperation is a bistatic sonar configuration. Typically the surface ship will transmit with an active sonar, with passive receiver antennas located on (or towed behind) one or more AUVs.

The cooperation between an AUV and a manned military submarine also offers interesting possibilities. The AUV may be used as a forward sensor platform to avoid exposure of the submarine. As another example, the AUV may act as a decoy, using sonar to imitate a real submarine's acoustic signature, thus deceiving the opponent.

Planning of cooperative missions can primarily be divided in two areas. If the mission can be split into non-overlapping tasks each involving only a single vehicle, then these tasks can be planned separately. Each vehicle in the system will have one or more of these tasks, and will execute them independently of the others. The only procedure involving actual cooperation will be the distribution of tasks. However, if the mission can not be split into non-overlapping tasks, the planning will become very complex. Further, a high level of active cooperation between the vehicles will require a high level of communication. Long-range, high-bandwidth, networkable underwater communication is technically difficult to achieve, and also undesirable if the vehicles should avoid being detected. The level of

cooperation is thus highly dependent on the mission, but it is evident that several missions can greatly benefit from using cooperating systems.

3.7 Implementation in the HUGIN AUV

Goal driven mission management systems produce a mission plan based on the overall mission objectives together with relevant constraints and prior knowledge of the mission area. Such a system is being developed for the HUGIN AUV. A hybrid control architecture, as described in Section 3.1, is used. The deliberate layer has a hierarchical structure, dividing each task into smaller subtasks until the subtasks can be implemented by the lower, reactive layer. This gives much flexibility in placing the line between the deliberate and the reactive part, and the solution fits well into the existing HUGIN system.

Using existing well-proven software as a starting point reduces the effort required for implementation and testing. It also facilitates a stepwise development approach, where new features can be tested at sea early in the process. While extensive testing can be performed in simulations, there is no substitute for testing AUV subsystems in their natural environment – the sea. This has been one of the main principles of the HUGIN development programme.

A framework for autonomy has been designed and implemented, and is being integrated with the existing HUGIN control and mission management system. Among the first features to utilize this framework are an advanced anti-collision system, automated surfacing for GPS position updates (controlled by a variety of constraints), and adaptive vehicle track parameters and sensor settings to optimize sensor performance. In this way, incremental steps will be taken towards a complete goal driven mission management system.

Automated mission planning will also be beneficial for mission preparation, simplifying the work of the operator and reducing the risk of mission failure due to human errors.

4. Discussion

Increasing the autonomy of AUVs will open many new markets for such vehicles – but it should also provide substantial benefits to current users: Better power sources facilitate longer endurance and/or more power-hungry sensors. Increased navigation autonomy relaxes the requirement for USBL positioning from a surface vessel, the frequency of GPS surface fixes etc. Perhaps most importantly, increased decision autonomy (including sustainability) will increase the probability of successfull completion of missions in all environments, and will also facilitate new missions and new modes of operation.

A shift from a manually programmed mission plan to a computer-generated plan based on higher-level operator input will also provide other benefits. Although graphical planning and simulation aids are used extensively with current AUVs, human errors in the planning phase still account for a significant portion of unsuccessful AUV missions. Increasing the automation in the mission planning process and elevating the human operator to a defining and supervisory role will eliminate certain types of errors.

The combined effect of increased energy, navigation and decision autonomy in AUVs will be seen over the next decade. The conservative nature of many current and potential users of AUVs dictates a stepwise adoption of new technology. However, even fairly modest, incremental improvements will facilitate new applications.

5. References

Arkin, R. C. (1998). *Behavior-based robotics*, The MIT Press, Cambridge, USA

Bellettini, A. & Pinto, M. A. (2002). Theoretical accuracy of synthetic aperture sonar micronavigation using a displaced phase-center antenna. *IEEE Journal of Oceanic Engineering*, vol. 27 no. 4, pp. 780-789

Bergman, N. ; Jung, L. & Gustafsson, F. (1999). Terrain navigation using Bayesian statistics, *IEEE Control System Magazine*, Vol. 19, No. 3, 1999, pp. 33-40

Bonabeau, E. ; Dorigo, M. & Theraulaz, G. (1999). *Swarm intelligence : from natural to artificial systems*. Oxford University Press. New York, NY, USA

Bourgeois, B. S.; Martinez, A. B.; Alleman, P. J.; Cheramie, J. J. & Gravley, J. M. (1999). Autonomous bathymetry survey system, *IEEE Journal of Oceanic Engineering,* Vol 24, No. 4, 1999, pp. 414 – 423

Brooks, R. A. (1986). A robust layered control system for a mobile robot, *IEEE Journal of robotics and automation,* vol. 2, No. 1, March 1986, pp. 14-23

Brutzman, D.; Healey, T.; Marco, D. & McGhee, B. (1998). The Phoenix autonomous underwater vehicle, *Artificial intelligence and mobile robots.* Kortenkamp, D.; Bonasso, R. P.; Murphy, R. (Ed.), pp. 323-360, The MIT Press, Cambridge, USA

Carreras, M.; Batlle, J. ; Ridao, P. & Roberts, G. N. (2000). An overview on behaviour-based methods for AUV control, *Proceedings of MCMC2000, 5th IFAC Conference on Manoeuvring and Control of Marine Crafts,* Aalborg, Denmark, August 2000

Choset, H. (2001). Coverage for robotics – A survey of recent results, *Annals of Mathematics and Artificial Intelligence,* Vol 31, pp. 113-126

Dijkstra, E. W. (1959). A note on two problems in connection with graphs, *Numerische Mathematik,* Vol 1, 1959, pp. 269-271

Fossum, T. ; Hagen, P. E. & Hansen, R. E. (2008). HISAS 1030 : The next generation mine hunting sonar for AUVs. *UDT Europe 2008 Conference Proceedings*, Glasgow, UK, June 2008

Gade, K. (2004). NavLab, a Generic Simulation and Post-processing Tool for Navigation, *European Journal of Navigation*, vol 2 no 4, November 2004

Gat, E. (1998). Three-layer architectures, *Artificial intelligence and mobile robots.* Kortenkamp, D. ; Bonasso, R. P. & Murphy, R. (Ed.), pp. 195-210, The MIT Press, Cambridge, USA

Golden, J. (1980). Terrain Contour matching(TERCOM) : A cruise missile guidance aid, In : *Image Processing for Missile Guidance*, Wiener, T. (Ed.), The Society of Photo-Optical Engineeers, Vol. 238, 1980, pp. 10-18

Hagen, O. K. & Hagen P. E. (2000). Terrain referenced integrated navigation system for underwater vehicles, *SACLANTCEN Conference Proceedings* CP-46, Bovio, E. Tyce, R. and Schmidt, H. (Ed.), pp. 171-180, NATO SACLANT Undersea Research Centre, La Spezia, Italy

Hagen, O. K. (2006). TerrLab – a generic simulation and post-processing tool for terrain referenced navigation, *Proceedings of Oceans 2006 MTS/IEEE*, Boston, MA, USA, September 2006

Hagen, P. E.; Hansen, R. E.; Gade, K. & Hammerstad, E. (2001). Interferometric Synthetic Aperture Sonar for AUV Based Mine Hunting: The SENSOTEK project, *Proceedings of Unmanned Systems 2001*, Baltimore, MD, USA, July-August 2001

Hagen, P. E.; Midtgaard, Ø. & Hasvold, Ø. (2007). Making AUVs Truly Autonomous, *Proceedings of Oceans 2007 MTS/IEEE*, Vancouver, BC, Canada, October 2007

Hansen, R. E.; Sæbø, T. O.; Gade, K. & Chapman, S. (2003). Signal Processing for AUV Based Interferometric Synthetic Aperture Sonar, *Proceedings of Oceans 2003 MTS/IEEE*, San Diego, CA, USA, September 2003

Hasvold, Ø.; Størkersen, N.; Forseth, S. & Lian, T. (2006). Power sources for autonomous underwater vehicles, *Journal of Power Sources*, vol. 162 no. 2, pp. 935-942, November 2006

Hegrenæs, Ø.; Hallingstad, O. & Gade, K. (2007). Towards Model-Aided Navigation of Underwater Vehicles. *Modeling, Identification and Control*, vol. 28, no. 4, October 2007, pp. 113-123

Hegrenæs, Ø.; Berglund, E. & Hallingstad, O. (2008). Model-Aided Inertial Navigation for Underwater Vehicles, *Proceedings of IEEE International Conference on Robotics and Automation 2008 (ICRA-08)*, Pasadena, CA, USA, May 2008, pp. 1069-1076

Hert, S.; Tiwari, S. & Lumensky, V. (1996). A Terrain-Covering Algorithm for an AUV, *Autonomous Robots*, Vol. 3, No. 2, 1996, pp. 91-119

Hostetler, L. (1978). Optimal terrain.aided navigation systems, *AIAA Guidance and Control Conference,* Palo Alto, CA, USA, 1978

Huang, W. H. (2001). Optimal line-sweep-based decompositions for coverage algorithms, *Proceedings of the 2001 IEEE International conference on Robotics and Automation,* pp. 27-32, Seoul, Korea 2001

Jalving, B. ; Gade, K. ; Hagen, O. K. & Vestgård, K (2003). A Toolbox of Aiding Techniques for the HUGIN AUV Integrated Inertial Navigation System, *Proceedings of Oceans 2003 MTS/IEEE*, San Diego, CA, USA, September 2003

Jalving, B. ; Vestgård, K.; Faugstadmo, J. E. ; Hegrenæs, Ø.; Engelhardtsen, Ø. & Hyland, B. (2008). Payload sensors, navigation and risk reduction for AUV under ice surveys, *Proceedings of Oceans 2008 MTS/IEEE*, Quebec, QC, Canada, September 2008

LaValle, S. M. (2006). *Planning Algorithms,* Cambridge University Press, ISBN 0-521-86205-9, New York, USA

Midtgaard, Ø.; Jalving, B. & Hagen, P.E. (2006), *Initial design of anti-collision system for HUGIN AUV*, FFI/RAPPORT 2006/01906 (IN CONFIDENCE), 2006

Nocedal, J. & Wright, S. J. (1999). *Numerical Optimization*, Springer-Verlag, ISBN 0-387-98793-2, New York, USA

Ridao, R.; Yuh, J.; Battle, J. & Sugihara, K. (2000). On AUV control architecture. *Proceedings of International Conference on Intelligent Robots and Systems (IROS 2000)*, Takamatsu, Japan

Russell, S. & Norvig, P. (2003). *Artificial intelligence – a modern approach, Ch 25: Robotics*. Second edition, Prentice Hall, New Jersey, USA.

Tan, C. S.; Sutton R. & Chudley J. (2004). Collision avoidance systems for autonomous underwater vehicles (Parts A and B), *Journal of Marine Science and Environment,* No. C2 , November 2004, pp. 39-62

Valavanis, K. P. ; Gracanin, D.; Matijasevic, M.; Kolluru, R. & Demetriou, G. A. (1998). Control architectures for autonomous underwater vehicles. *IEEE Control Systems Magazine*, Vol. 17, No. 6

9

Nonlinear Control Methodologies for Tracking Configuration Variables

Poorya Haghi
Amirkabir University of Technology
Iran

1. Introduction

The problem of designing an accurate and reliable control for an Autonomous Underwater Vehicle (AUV), which is being subjected to environmental disturbances as well as configuration related changes, is critical in order to accomplish a successful mission. Any real-world problem solving system must deal with the issue of uncertainty, since the system's knowledge of the world is always incomplete, imprecise, and uncertain. This situation is aggravated for an AUV, due to the complex oceanic environment, and the inevitable noise of the sensory system.
Some major facts that contribute to the difficulty of the underwater vehicle control are:

- the dynamic behavior of the vehicle is highly nonlinear,
- hydrodynamic coefficients cannot be easily obtained, hence making up uncertainties in the model knowledge,
- the vehicle main body can be disturbed due to the ocean currents and vehicle motion.

Therefore, it is difficult to obtain high performance by using the conventional control strategies. The control system should be able to learn and adapt itself to the changes in the dynamics of the vehicle and its environment.
Many control methods have been proposed by researchers during the last decade, and there still exists a trend towards finding a better control law to achieve exponential stability while accounting for environmental changes and vehicle uncertainties. Focusing on the low level motion control of AUVs, most of the proposed control schemes take into account the uncertainty in the model by resorting to an adaptive strategy ((Corradini & Orlando, 1997), (Fossen & Sagatun, 1991a) and (Narasimhan & Singh, 2006)), or a robust approach ((Marco & Healey, 2001) and (Healey & Lienard, 1993)). In (Healey & Lienard, 1993) an estimation of the dynamic parameters of the vehicle NPS AUV *Phoenix* is also provided. Other relevant works on the adaptive and robust control of underwater vehicles are (Cristi & Healey, 1989), and (Cristi et al., 1990). (Leonard & Krishnaprasad, 1994) considers the control of an AUV in the event of an actuator failure. Experimental results on underwater vehicle control have been addressed by many researchers (e.g. see (Antonelli et al., 1999), (Antonelli et al., 2001), and (Zhao & Yuh, 2005)). An overview of control techniques for AUVs is reported in (Fossen, 1994).
The aim of this chapter is to design a control system that would achieve perfect tracking for all configuration variables (e.g. sway and yaw motions) for any desired trajectory. To this end, we present the application of nonlinear control methods to an AUV that would lead to

a successful uncertainty management, while accounting for the effect of saturation: an unwanted implementation problem which is seldom addressed by researchers.
Three control methods are presented and applied to a two-dimensional model of an AUV, and their capabilities to cope with the issues of parameter uncertainties and environmental disturbances are studied and compared. The considered model is a nonlinear multi-input multi-output (MIMO) system, therefore we intend to shed a light on the complexities encountered when dealing with such systems. This model also serves as an example, and helps clarify the application of the given methods. All the methods presented, guarantee perfect tracking for all configuration variables of the system. The performance of the presented methods, are compared via simulation studies.
We begin by designing a control law using the computed torque control method. Although simple in design, the stability achieved by this method is sensitive to parameter variations and noise of the sensory system. Moreover, the maximum amount of disturbance waves that can be conquered by this method is somewhat lower relative to the other methods given here. Next we present the adaptive approach to computed torque control method. It will be shown that this method can withstand much higher values of disturbance waves and remain stable. Furthermore, parameter variations are compensated through an adaptation law. The third method presented, is the suction control method in which we employ the concepts of *sliding surfaces*, and *boundary layers*. This method, being robust in nature, achieves an optimal trade-off between control bandwidth and tracking precision. Compared to the computed torque control method, this method has improved performance with a more tractable controller design. Finally, the effect of saturation is studied through a novel approach, by considering the desired trajectory. A condition is derived under which saturation will not occur. The chapter will be closed by proposing topics for further research.

2. Nonlinear control methodologies

All physical systems are nonlinear to some extent. Several inherent properties of linear systems which greatly simplify the solution for this class of systems, are not valid for nonlinear systems (Shinner, 1998). The fact that nonlinear systems do not have these properties further complicates their analysis. Moreover, nonlinearities usually appear multiplied with physical constants, often poorly known or dependent on the slowly changing environment, thereby increasing the complexities. Therefore, it is important that one acquires a facility for analyzing control systems with varying degrees of nonlinearity.
This section introduces three nonlinear control methods for tracking purposes. To maintain generality, we consider a general dynamic model of the form

$$\mathbf{T} = \mathbf{H(q)}\ddot{\mathbf{q}} + \mathbf{C(q,\dot{q})}\dot{\mathbf{q}} + \mathbf{G(q)} \tag{1}$$

that can represent the dynamic model of numerous mechanical systems such as robotic vehicles, robot manipulators, etc, where $\mathbf{H(q)}$ is an $n \times n$ matrix, representing mass matrix or inertia matrix (including added mass for underwater vehicles), $\mathbf{C(q,\dot{q})}$ represents the matrix of Coriolis and centripetal terms (including added mass for underwater vehicles), and $\mathbf{G(q)}$ is the vector of gravitational forces and moments. For the case of underwater vehicles, which is the main concern of this chapter, the term $\mathbf{C(q,\dot{q})}$ will also represent the hydrodynamic damping and lift matrix. The methods given in this section, will be applied to an underwater vehicle model in section 3.

2.1 Computed torque control method

This section presents a nonlinear control method, apparently first proposed in (Paul, 1972) and named the *computed torque method* in (Markiewicz, 1973) and (Bejczy, 1974). This method is based on using the dynamic model of the system in the control law formulation. Such a control formulation yields a controller that suppresses disturbances and tracks desired trajectories uniformly in all configurations of the system (Craig, 1988).

Suppose that the system's dynamics is governed by Eq. (1). The control objective is to track a desired trajectory $\mathbf{q_d}$. Such a trajectory may be preplanned by several well-known schemes (Craig, 1989). We define a tracking error $\tilde{\mathbf{q}}$

$$\tilde{\mathbf{q}} = \mathbf{q_d} - \mathbf{q}, \tag{2}$$

and make the following proposition.

Proposition 2.1 *The control law*

$$\mathbf{T} = \mathbf{H(q)u} + \mathbf{C(q,\dot{q})\dot{q}} + \mathbf{G(q)} \tag{3}$$

can track any desired trajectory $\mathbf{q_d}$, *as long as the matrices* $\mathbf{H}$, $\mathbf{C}$, *and* $\mathbf{G}$ *are known to the designer. The servo law,* $\mathbf{u}$, *is given by*

$$\mathbf{u} = \mathbf{q_d} + \mathbf{K_v q} + \mathbf{K_p q}, \tag{4}$$

where $\mathbf{K_v}$ *and* $\mathbf{K_p}$ *are called servo gain matrices.*

Proof.

Substituting the proposed control law into the equation of motion, Eq. (1), we obtain

$$\mathbf{H(q)\ddot{q}} + \mathbf{C(q,\dot{q})\dot{q}} + \mathbf{G(q)} = \mathbf{H(q)(q_d} + \mathbf{K_v q} + \mathbf{K_p q)} + \mathbf{C(q,\dot{q})\dot{q}} + \mathbf{G(q)},$$

which yields the following error dynamics

$$\ddot{\tilde{\mathbf{q}}} + \mathbf{K_v}\dot{\tilde{\mathbf{q}}} + \mathbf{K_p}\tilde{\mathbf{q}} = \mathbf{0}.$$

A proper choice of the servo gain matrices will lead to a stable error dynamics. One such example is given by the following matrices

$$\mathbf{K_p} = \mathrm{diag}[\lambda_1^2, \lambda_2^2, \ldots, \lambda_n^2] \tag{5}$$

$$\mathbf{K_v} = \mathrm{diag}[2\lambda_1, 2\lambda_2, \ldots, 2\lambda_n], \tag{6}$$

where λ_i are adjustable design parameters. ▪▪

It can be seen that this control formulation exhibits perfect tracking for any desired trajectory. But this desired performance is based on the underlying assumption that the values of parameters appearing in the dynamic model in the control law match the parameters of the actual system, which makes the implementations of the computed torque control less than ideal due to the inevitable uncertainties of the system, e.g. resulting from unknown hydrodynamic coefficients. In the existence of uncertainties, the control law (3) must be modified to

$$\mathbf{T} = \hat{\mathbf{H}}\mathbf{(q)u} + \hat{\mathbf{C}}\mathbf{(q,\dot{q})\dot{q}} + \hat{\mathbf{G}}\mathbf{(q)}, \tag{7}$$

where $\widehat{[\cdot]}$ denotes the estimation of matrix $[\cdot]$. One can show that substitution of the above control law into the equation of motion will lead to the following error dynamics

$$\ddot{\tilde{\mathbf{q}}} + \mathbf{K}_v\dot{\tilde{\mathbf{q}}} + \mathbf{K}_p\tilde{\mathbf{q}} = \hat{\mathbf{H}}^{-1}\tilde{\mathbf{T}}, \tag{8}$$

where $\tilde{\mathbf{T}} = \tilde{\mathbf{H}}\ddot{\mathbf{q}} + \tilde{\mathbf{C}}\dot{\mathbf{q}} + \tilde{\mathbf{G}}$, and the tilde matrices are defined by $\widetilde{[\cdot]} = [\cdot] - \widehat{[\cdot]}$. Since the right hand side of the error dynamics is not zero anymore, this method becomes inefficient in the presence of uncertainties. This problem is conquered by the adaptive counterpart of the computed torque control method.

2.2 Adaptive computed torque control method

In this section, we introduce the adaptive computed torque control method, and derive an adaptation law to estimate the unknown parameters. The control of nonlinear systems with unknown parameters is traditionally approached as an adaptive control problem. Adaptive control is one of the ideas conceived in the 1950's which has firmly remained in the mainstream of research activity with hundreds of papers and several books published every year. One reason for the rapid growth and continuing popularity of adaptive control is its clearly defined goal: to control plants with unknown parameters. Adaptive control has been most successful for plant models in which the unknown parameters appear linearly. But in many mechanical systems, the unknown parameters appear in a nonlinear manner. For such systems we define *parameter functions*, $\mathbf{P}$, such that the system have a linear relationship with respect to these parameter functions. Fortunately, such a *linear parameterization* can be achieved in most situations of practical interest (Kristic et al., 1995). We only consider such systems throughout this work.

In the linear parameterization process, we partition the system into a **model-based portion** and a **servo portion**. The result is that the system's parameters appear only in the model-based portion, and the servo portion is independent of these parameters. This partitioning involves the determination of parameter functions $\mathbf{P}$, such that the error dynamics is linear in the parameter functions. When this is possible, one can write

$$\tilde{\mathbf{T}} = \tilde{\mathbf{H}}(\mathbf{q})\ddot{\mathbf{q}} + \tilde{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \tilde{\mathbf{G}}(\mathbf{q}) \equiv \mathbf{W}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\tilde{\mathbf{P}}, \tag{9}$$

where $\mathbf{W}$ is a $n \times k$ matrix, called the regression matrix, and $\tilde{\mathbf{P}}$ is a $k \times 1$ vector, representing the parameter function estimation errors and is defined by $\tilde{\mathbf{P}} = \mathbf{P} - \hat{\mathbf{P}}$.

Once the parameterization process is done successfully, one can employ the following adaptation law to estimate the parameter functions.

Proposition 2.2 *For a system with either constant or slowly varying unknown parameters, the adaptation law*

$$\dot{\hat{\mathbf{P}}} = \mathbf{\Gamma}\mathbf{W}^{\mathrm{T}}\widehat{\mathbf{H}}^{-\mathrm{T}}\mathbf{Y}, \tag{10}$$

estimates the parameter functions, such that the error dynamics of Eq. (8) becomes stable. Definitions of $\mathbf{\Gamma}$ *and* $\mathbf{Y}$ *are given in the following proof.*

Proof.

The error dynamics is given by Eq. (8). Substituting for $\tilde{\mathbf{T}}$ from the linear parameterization law, Eq. (9), we have

$$\ddot{\tilde{\mathbf{q}}} + \mathbf{K}_v\dot{\tilde{\mathbf{q}}} + \mathbf{K}_p\tilde{\mathbf{q}} = \hat{\mathbf{H}}^{-1}\mathbf{W}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\tilde{\mathbf{P}}, \tag{11}$$

The aim of the adaptation law is to estimate the parameter functions $\mathbf{P}$, so as to make the right hand side of the above equation approach zero, i.e. by making $\tilde{\mathbf{P}}$ approach zero. One can write Eq. (11) in state space form by defining the state vector $\mathbf{X}$ as

$$\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n]^{\mathrm{T}}, \qquad \mathbf{X}_i \triangleq [\tilde{q}_i, \dot{\tilde{q}}_i]^T,$$

and the output vector $\mathbf{Y}$ as

$$\mathbf{Y} \triangleq \dot{\tilde{\mathbf{q}}} + \mathbf{\Phi}\,\tilde{\mathbf{q}}, \qquad \mathbf{\Phi} = \mathrm{diag}[\phi_1, \phi_2, \ldots, \phi_n],$$

where $\mathbf{\Phi}$ is the filtering matrix, and $\mathbf{Y}$ represents the vector of filtered errors. The values of ϕ_j must be chosen such that the transfer function

$$\frac{s + \phi_j}{s^2 + K_{vj}s + K_{pj}}$$

is *strictly positive real* (SPR)[1]. Therefore

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}\left(\hat{\mathbf{H}}^{-1}\mathbf{W}\tilde{\mathbf{P}}\right)$$

$$\mathbf{Y} = \mathbf{C}\mathbf{X}.$$

Having written the error dynamics in state space form, we employ a Lyapunov-based approach to derive the adaptation law. Consider the following Lyapunov candidate,

$$V = \mathbf{X}^{\mathrm{T}}\mathbb{P}\mathbf{X} + \tilde{\mathbf{P}}^{\mathrm{T}}\mathbf{\Gamma}^{-1}\tilde{\mathbf{P}}, \tag{12}$$

where $\mathbb{P}$ is a positive definite matrix, and $\mathbf{\Gamma} = \mathrm{diag}[\gamma_1, \gamma_2, \ldots, \gamma_r]$ with $\gamma_i > 0$. Taking the time derivative of (12) yields

$$\dot{V} = \dot{\mathbf{X}}^{\mathrm{T}}\mathbb{P}\mathbf{X} + \mathbf{X}^{\mathrm{T}}\mathbb{P}\dot{\mathbf{X}} + 2\tilde{\mathbf{p}}^{\mathrm{T}}\mathbf{\Gamma}^{-1}\dot{\tilde{\mathbf{p}}}. \tag{13}$$

Substitution of the state space equations of error dynamics into (13) results

$$\dot{V} = 2\tilde{\mathbf{p}}^{\mathrm{T}}\left[\mathbf{\Gamma}^{-1}\dot{\tilde{\mathbf{p}}} + \mathbf{W}^{\mathrm{T}}\hat{\mathbf{H}}^{-\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbb{P}\mathbf{X}\right] + \mathbf{X}^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbb{P} + \mathbb{P}\mathbf{A})\mathbf{X}. \tag{14}$$

This equation can further be simplified, by adopting the following lemma.

Lemma 2.1 (Kalman-Yakubovich-Popov) *Consider a controllable linear time-invariant system*

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}\mathbf{u}$$

---

[1]A transfer function $h(p)$ is *positive real* if

$$Re[h(p)] \geq 0 \quad \textit{for all} \quad Re[p] \geq 0$$

It is *strictly positive real* if $h(p-\varepsilon)$ is positive real for some $\varepsilon > 0$.

$$y = \mathbf{cx}.$$

The transfer function $h(p) = \mathbf{c}[p\mathbf{I} - \mathbf{A}]^{-1}\mathbf{b}$ *is SPR if, and only if, there exist positive definite matrices* $\mathbb{P}$ *and* $\mathbb{Q}$ *such that*

$$\mathbf{A}^{\mathbf{T}}\mathbb{P} + \mathbb{P}\mathbf{A} = -\mathbb{Q}$$

$$\mathbb{P}\mathbf{b} = \mathbf{c}^{\mathbf{T}}.$$

According to the above lemma, one can write $(\mathbf{A}^{\mathbf{T}}\mathbb{P} + \mathbb{P}\mathbf{A}) = -\mathbb{Q}$ in Eq. (14). The adaptation law is found by setting the first term on the right side of (14) equal to zero

$$2\tilde{\mathbf{p}}^{\mathbf{T}}\left[\mathbf{\Gamma}^{-1}\dot{\tilde{\mathbf{p}}} + \mathbf{W}^{\mathbf{T}}\left(\hat{\mathbf{H}}^{-1}\right)^{\mathbf{T}}\mathbf{B}^{\mathbf{T}}\mathbb{P}\mathbf{X}\right] = 0,$$

Rearranging the above equation and noting that $\dot{\tilde{\mathbf{p}}} = -\dot{\hat{\mathbf{p}}}$ and using Lemma 2.1, the adaptation law is found as

$$\dot{\hat{\mathbf{p}}} = \mathbf{\Gamma}\mathbf{W}^{\mathbf{T}}\hat{\mathbf{H}}^{-\mathbf{T}}\mathbf{Y},$$

and Eq. (14) will become

$$\dot{V} = -\mathbf{X}^{\mathbf{T}}\mathbb{Q}\mathbf{X},$$

which is a stable Lyapunov function. ▪▪

Even though $\mathbf{H}^{-1}$ always exists in a physical problem, a vigilant reader might question the existence of $\hat{\mathbf{H}}^{-1}$. It is shown in (Craig, 1988), that $\hat{\mathbf{H}}$ will remain positive definite and invertible, if we ensure that all parameters remain within a sufficiently small range near the actual parameter value. See (Craig, 1988) for the details of how this is done.

2.3 Suction control

One major approach to dealing with model uncertainty is the robust control. Broadly speaking, robustness is a property which guarantees that essential functions of the designed system are maintained under adverse conditions in which the model no longer accurately reflects reality. In modeling for robust control design, an exactly known nominal plant is accompanied by a description of plant uncertainty, that is, a characterization of how the true plant might differ from the nominal one. This uncertainty is then taken into account during the design process (Freeman & Kokotovic, 1996).

For simplicity, we explain the method for a single-input system. The extension to multi-input systems is straight forward, as will be illustrated in the AUV example. A more detailed discussion of this method is given by (Slotine, 1985), (Slotine & Sastry, 1983), and (Slotine & Li, 1991).

Consider the dynamic system

$$x^{(n)}(t) = f(\mathbf{X};t) + b(\mathbf{X};t)u(t), \tag{15}$$

where $u(t)$ is the control input and $\mathbf{X} = [x, \dot{x}, \ldots, x^{(n-1)}]^T$ is the state vector. It is assumed that the generally nonlinear function $f(\mathbf{X};t)$ is not exactly known, but the extent of imprecision on f is upper-bounded by a known continuous function of $\mathbf{X}$ and t. Similarly the control gain $b(\mathbf{X};t)$ is not exactly known, but is of constant sign and is bounded by known continuous functions of $\mathbf{X}$ and t. The control problem is to track the desired trajectory

$\mathbf{X}_d = [x_d, \dot{x}_d, \ldots, x_d^{(n-1)}]^T$ in the presence of model imprecisions on f and b. Defining the tracking error as usual, $\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{X}_d$, we assume that

$$\tilde{\mathbf{X}}\big|_{t=0} = 0. \tag{16}$$

A time-varying sliding surface $S(t)$ is defined in the state space $\mathbb{R}^n$ as $S(t) : s(\mathbf{X};t) = 0$, with

$$s(\mathbf{X};t) \triangleq \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x}, \qquad \lambda > 0, \tag{17}$$

where λ is a positive constant. Given the initial condition (16), the problem of tracking X_d is equivalent to that of remaining on the surface $S(t)$ for all $t > 0$. Now a sufficient condition for such positive invariance of $S(t)$ is to choose the control law u of Eq. (15) such that outside of sliding condition $S(t)$, the following holds:

$$\frac{1}{2}\frac{d}{dt}s^2(\mathbf{X};t) \le -k\,|s|, \tag{18}$$

where k is a positive constant. Sliding condition (18) constraints state trajectories to point toward the sliding surface $S(t)$. Geometrically, it looks like the trajectories are sliding down $S(t)$ to reach the desired state. Satisfying Eq. (18) guarantees that if condition (16) is not exactly verified, the surface $S(t)$ will nonetheless be reached in a finite time, while definition (17) then guarantees that $\tilde{\mathbf{X}} \to 0$ as $t \to \infty$ (Slotine, 1985).

The controller design procedure in the suction control method, consists of two steps. First, a feedback control law u is selected so as to verify sliding condition (18). Such a control law is discontinuous across the surface, which leads to control *chattering*. Chattering is undesirable in practice because it involves high control activity and further may excite high-frequency dynamics neglected in the course of modeling. Thus in a second step, discontinuous control law u is suitably smoothed to achieve an optimal trade-off between control bandwidth and tracking precision. While the first step accounts for parametric uncertainty, the second step achieves robustness to high-frequency unmodeled dynamics. Construction of a control law to verify the sliding condition (18) is straight forward, and will be illustrated in section 3.4 through an example.

3. A two-dimensional model of a MIMO AUV

In this section the problem of tracking the configuration variables (position and attitude) of an AUV in the horizontal plane is considered. Two rudders in front and rear side of the vehicle are used as control inputs, and the methods of previous section are applied. A schematic diagram of the system under consideration is shown in Fig. 1.

3.1 Dynamic modeling

The dynamic behavior of an underwater vehicle is described through Newton's laws of linear and angular momentum. The equations of motion of such vehicles are highly nonlinear, time-varying and coupled due to hydrodynamic added mass, lift, drag, Coriolis and centripetal forces, which are acting on the vehicle and generally include uncertainties (Fossen & Sagatun, 1991b). Detailed discussions on modeling and system identification techniques are given in (Fossen, 1994) and (Goheen & Jefferys, 1990).

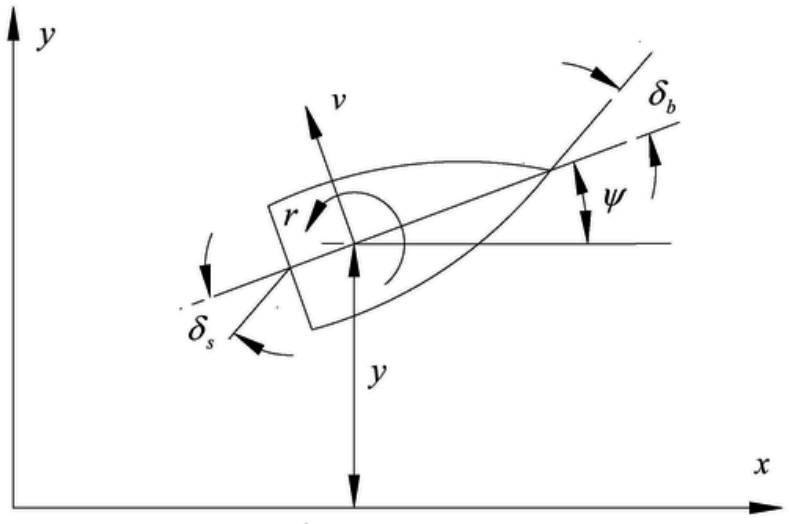


Fig. 1. Geometry and axes definition of an AUV.

It is convenient to write the equations of motion in accordance with the Society of National Architects and Marine Engineers (SNAME, 1950). Restricting our attention to the horizontal plane, the mathematical model consists of the nonlinear sway (translational motion with respect to the vehicle longitudinal axis) and yaw (rotational motion with respect to the vertical axis) equations of motion. According to (Haghi et al., 2007), these equations are described by

$$\dot{v}[m - Y_{\dot{v}}] + \dot{r}[mx_G - Y_{\dot{r}}] = Y_{\delta_s}\delta_s u^2 + Y_{\delta_b}\delta_b u^2 - d_1(v,r) + Y_v uv + (Y_r - m)ur \tag{19}$$

$$\dot{v}[mx_G - N_{\dot{v}}] + \dot{r}[I_z - N_{\dot{r}}] = N_{\delta_s}\delta_s u^2 + N_{\delta_b}\delta_b u^2 - d_2(v,r) + N_v uv + (N_r - mx_G)ur, \tag{20}$$

where $d_1(v,r)$ and $d_2(v,r)$ are defined as

$$d_1(v,r) \triangleq \frac{\rho}{2}\int_{tail}^{nose} C_{Dy}\, h(\xi)\frac{(v+\xi r)^3}{|v+\xi r|}d\xi$$

$$d_2(v,r) \triangleq \frac{\rho}{2}\int_{tail}^{nose} C_{Dy}\, h(\xi)\frac{(v+\xi r)^3}{|v+\xi r|}\xi d\xi.$$

Equations (19) and (20), along with the expressions for the vehicle yaw rate and the inertial position rates, describe the complete model of the vehicle. For control purposes it is convenient to solve Eqs. (19) and (20) for $\dot{v}$ and $\dot{r}$. Therefore the complete set of equations of motion is

$$\dot{v} = a_{11}uv + a_{12}ur + d_v(v,r) + b_{11}u^2\delta_s + b_{12}u^2\delta_b \tag{21}$$

$$\dot{r} = a_{21}uv + a_{22}ur + d_r(v,r) + b_{21}u^2\delta_s + b_{22}u^2\delta_b \tag{22}$$

$$\dot{\psi} = r \tag{23}$$

$$\dot{x} = u\cos\psi - v\sin\psi \tag{24}$$

$$\dot{y} = u\sin\psi + v\cos\psi, \tag{25}$$

where a_{ij}, b_{ij} and c_i are the related coefficients that appear when solving (19) and (20) for $\dot{v}$ and $\dot{r}$.

During regular cruising, the drag related terms $d_v(v,r)$ and $d_r(v,r)$ are small, and can be neglected (Yuh, 1995). Note that all the parameters a_{ij} and b_{ij}, include at least two hydrodynamic coefficients, such as $Y_{\dot{v}}, Y_{\dot{r}}, N_{\dot{v}}, N_{\dot{r}}, \ldots$; hence uncertainties. In the proceeding sections, we apply the nonlinear control methods of the previous section to this model. Our goal is to achieve perfect tracking for both sway and yaw motions of the vehicle.

3.2 Computed torque control method

Suppose that it is desired that the sway motion of the vehicle tracks the preplanned trajectory y_d, and that the yaw motion of the vehicle tracks the preplanned trajectory ψ_d. Let the tracking errors be defined by

$$\tilde{y} = y - y_d \tag{26}$$

$$\tilde{\psi} = \psi - \psi_d. \tag{27}$$

The control law is given by Eqs. (3) and (4). One can observe that Eq. (3) is obtained by replacing the acceleration term of the equations of motion, $\ddot{\mathbf{q}}$, by the servo law $\mathbf{u}$. Since this process involves the acceleration terms, we take the time derivative of Eqs. (23) and (25), and substitute (21) and (22) into the results. Therefore

$$\ddot{y} = u\dot{\psi}\cos\psi + (a_{11}uv + a_{12}u\dot{\psi} + b_{11}u^2\delta_s + b_{12}u^2\delta_b)\cos\psi - v\dot{\psi}\sin\psi \tag{28}$$

$$\ddot{\psi} = a_{21}uv + a_{22}ur + b_{21}u^2\delta_s + b_{22}u^2\delta_b. \tag{29}$$

Next we replace $\ddot{y}$ with the servo law μ, and $\ddot{\psi}$ with the servo law ν, and solve these equations for the rudder deflections δ_s and δ_b to obtain the control law

$$\delta_s = \{b_{22}(\mu\sec\psi + v\dot{\psi}\tan\psi - u\dot{\psi} - a_{11}uv - a_{12}ur) - b_{12}(\nu - a_{21}uv - a_{22}ur)\}$$

$$/(b_{11}b_{22} - b_{21}b_{12})u^2 \tag{30}$$

$$\delta_b = -\{b_{21}(\mu\sec\psi + v\dot{\psi}\tan\psi - u\dot{\psi} - a_{11}uv - a_{12}ur) - b_{11}(\nu - a_{21}uv - a_{22}ur)\}$$

$$/(b_{11}b_{22} - b_{21}b_{12})u^2, \tag{31}$$

where the servo laws μ and ν are given according to Eqs. (4), (5), and (6)

$$\mu \triangleq \ddot{y}_d - 2\lambda_1\dot{\tilde{y}} - \lambda_1^2\tilde{y}$$

$$\nu \triangleq \ddot{\psi}_d - 2\lambda_2\dot{\tilde{\psi}} - \lambda_2^2\tilde{\psi}.$$

Note that the negative signs in the above equations, are due to the definition of tracking error in Eqs. (26) and (27) which differs in a minus sign from the definition of Eq. (2).

3.3 Adaptive computed torque control method

In this method, the control law estimates the unknown parameters. As stated before, all the parameters a_{ij} and b_{ij} comprise hydrodynamic uncertainties which must be estimated. On the other hand, the vehicle's forward velocity u is assumed to be constant, but subjected to changes from environment, and ocean currents. Thus all terms including u must also be estimated. But instead of estimating all the a_{ij} and b_{ij} terms, we define parameter functions, p_i , in a linear parameterization process. This process does not reveal a unique parameterization and the results depend on the way one defines p_i s. One can show that a possible parameterization of Eqs. (30) and (31) is given by

$$\delta_s = p_1\left(\frac{\mu + vr\sin\psi}{\cos\psi}\right) + p_2 v + p_3 r - p_4 \nu \tag{32}$$

$$\delta_b = -p_5\left(\frac{\mu + vr\sin\psi}{\cos\psi}\right) + p_6 v + p_7 r + p_8 \nu, \tag{33}$$

where

$$\begin{aligned}
p_1 &= \frac{b_{22}}{(b_{11}b_{22} - b_{21}b_{12})u^2} & p_2 &= \frac{b_{12}a_{21}u - b_{22}a_{11}u}{(b_{11}b_{22} - b_{21}b_{12})u^2} \\
p_3 &= \frac{b_{12}a_{22}u - b_{22}a_{12}u - b_{22}u}{(b_{11}b_{22} - b_{21}b_{12})u^2} & p_4 &= \frac{b_{12}}{(b_{11}b_{22} - b_{21}b_{12})u^2} \\
p_5 &= \frac{b_{21}}{(b_{11}b_{22} - b_{21}b_{12})u^2} & p_6 &= \frac{b_{21}a_{11}u - b_{11}a_{21}u}{(b_{11}b_{22} - b_{21}b_{12})u^2} \\
p_7 &= \frac{b_{21}a_{21}u - b_{11}a_{22}u + b_{21}u}{(b_{11}b_{22} - b_{21}b_{12})u^2} & p_8 &= \frac{b_{11}}{(b_{11}b_{22} - b_{21}b_{12})u^2}.
\end{aligned}$$

Since all the p_i s include uncertainties, the control law is modified as follows:

$$\delta_s = \hat{p}_1\left(\frac{\mu + vr\sin\psi}{\cos\psi}\right) + \hat{p}_2 v + \hat{p}_3 r - \hat{p}_4 \nu \tag{34}$$

$$\delta_b = -\hat{p}_5\left(\frac{\mu + vr\sin\psi}{\cos\psi}\right) + \hat{p}_6 v + \hat{p}_7 r + \hat{p}_8 \nu, \tag{35}$$

where $\hat{p}_i$ represents parameter estimations, and the servo signals μ and ν are defined as before. The next step is to derive the adaptation law.

Let the estimation error of parameters be $\tilde{p}_i = p_i - \hat{p}_i$. One can find the error dynamics by substituting (34) and (35) into the system dynamic equations. This results

$$\overbrace{\begin{bmatrix} \hat{p}_1 & -\hat{p}_4 \\ \hat{p}_5 & -\hat{p}_8 \end{bmatrix}}^{\hat{\mathbf{H}}} \begin{Bmatrix} \dfrac{\ddot{y}-\mu}{\cos\psi} \\ \ddot{\psi}-\nu \end{Bmatrix} = \underbrace{\begin{bmatrix} -\dfrac{\ddot{y}+vr\sin\psi}{\cos\psi} & -v & -r & \ddot{\psi} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\dfrac{\ddot{y}+vr\sin\psi}{\cos\psi} & v & r & \ddot{\psi} \end{bmatrix}}_{\mathbf{W}} \tilde{\mathbf{P}}, \tag{36}$$

or in matrix form

$$\begin{Bmatrix} \dfrac{\ddot{y}-\mu}{\cos\psi} \\ \ddot{\psi}-\nu \end{Bmatrix} = \hat{\mathbf{H}}^{-1}\mathbf{W}\tilde{\mathbf{P}}. \tag{37}$$

One can write Eq. (37) in state space form by defining the state vector X and the output vector $\mathbf{Y}$ as defined in section 2.2

$$\dot{\mathbf{X}} = \mathbf{AX} + \mathbf{B}\left(\hat{\mathbf{H}}^{-1}\mathbf{W}\tilde{\mathbf{p}}\right)$$

$$\mathbf{Y} = \dot{\tilde{\mathbf{N}}} + \mathbf{\Phi}\tilde{\mathbf{N}},$$

where $\mathbf{\Phi}$ =diag$[\phi_1, \phi_2]$, and $\mathbf{N} = [\tilde{y}, \tilde{\psi}]^T$. Having defined the necessary matrices, we can utilize the adaptation law given by Eq. (10):

$$\dot{\hat{\mathbf{P}}} = \mathbf{\Gamma}\mathbf{W}^{\mathbf{T}}\hat{\mathbf{H}}^{-\mathbf{T}}\mathbf{Y},$$

where $\hat{\mathbf{H}}$ and $\mathbf{W}$ are defined in Eq. (36).

3.4 Suction control

One can write the system's governing dynamics, in matrix form as:

$$\begin{Bmatrix} \ddot{y} \\ \ddot{\psi} \end{Bmatrix} = \begin{bmatrix} a_{11}u\cos\psi - r\sin\psi & u\cos\psi + a_{12}u\cos\psi \\ a_{12}u & a_{22}u \end{bmatrix} \begin{Bmatrix} v \\ r \end{Bmatrix}$$

$$+ \begin{bmatrix} b_{11}u^2\cos\psi & b_{12}u^2\cos\psi \\ b_{21}u^2 & b_{22}u^2 \end{bmatrix} \begin{Bmatrix} \delta_s \\ \delta_b \end{Bmatrix},$$

or in vector form as

$$\ddot{\mathbf{N}} = \mathbf{AX} + \mathbf{BU}.$$

With the objective of tracking desired trajectories, the sliding surfaces S_1 and S_2 are chosen as

$$s_1(y,t) = \dot{\tilde{y}} + \lambda_1\tilde{y} = 0$$

$$s_2(\psi,t) = \dot{\tilde{\psi}} + \lambda_2\tilde{\psi} = 0.$$

The terms s_1 and s_2 are also called *combined tracking errors*, and can be written as

$$s_1 = \dot{y} - \dot{y}_r$$

$$s_2 = \dot{\psi} - \dot{\psi}_r,$$

where

$$\dot{y}_r = \dot{y}_d - \lambda_1 \tilde{y}$$

$$\dot{\psi}_r = \dot{\psi}_d - \lambda_2 \tilde{\psi}$$

are the *reference signals*. For notational simplicity, we define the vectors $\mathbf{S} = [s_1, s_2]^T$, and $\dot{\mathbf{N}}_\mathbf{r} = [\dot{y}_r, \dot{\psi}_r]^T$. Considering the equality of the sliding condition (18), one can write

$$\dot{s}_i s_i = -k_i \mid s_i \mid,$$

or

$$\dot{s}_i = -k_i \, sgn(s_i).$$

Defining a vector $\mathbf{K}\text{sgn}(\mathbf{S})$, with the elements $k_i \, \text{sgn}(s_i)$, the sliding condition will be $\dot{\mathbf{S}} = -\mathbf{K}\text{sgn}(\mathbf{S})$. Differentiation of $\mathbf{S}$ yields

$$\dot{\mathbf{S}} = \ddot{\mathbf{N}} - \ddot{\mathbf{N}}_\mathbf{r}.$$

Substitution of the dynamic equation and solving the result for **U**, the control law is found to be

$$\mathbf{U} = \mathbf{B}^{-1}\left(\ddot{\mathbf{N}}_\mathbf{r} - \mathbf{AX} - \mathbf{K}\text{sgn}(\mathbf{S})\right).$$

The above control law is discontinuous across the sliding surface. Since the implementation of the associated control law is necessarily imperfect (for instance, in practice switching is not instantaneous), this leads to chattering. Chattering is undesirable in practice, since it involves high control activity and further may excite high frequency dynamics neglected in the course of modeling (such as unmodeled structural modes, neglected time-delays, and so on). Thus, in a second step, the discontinuous control law is suitably smoothed. This can be achieved by smoothing out the control discontinuity in a thin boundary layer neighboring the switching surface (Slotine & Li, 1991):

$$B(t) = \left\{\mathbf{x}, \mid s(\mathbf{x};t) \mid \leq \Phi\right\} \qquad \Phi > 0,$$

where Φ is the boundary layer thickness. In other words, outside of $B(t)$, we choose control law u as before (i.e. satisfying the sliding condition); all other trajectories starting inside $B(t=0)$ remain inside $B(t)$ for all $t \geq 0$. The mathematical operation for this to occur is to simply replace $\text{sgn}(s)$ with $\text{sat}\left(\frac{s}{\Phi}\right)$, with the saturation function defined as:

$$\begin{cases} \text{sat}(y) = y & \text{if } \mid y \mid \leq 1 \\ \text{sat}(y) = \text{sgn}(y) & \text{otherwise} \end{cases}$$

The control law derived by this method is robust in nature; therefore, insensitive to uncertainties and disturbances. One can adjust the robustness of the system by selecting

proper control gains. When the upper bounds and lower bounds of uncertainties and/or disturbances are known, one can include these bounds in the control law design, to assure the robustness of the system. See (Slotine & Sastry, 1983) for more information.

4. Simulations

For the purpose of simulations, the following numerical values have been used as in (Haghi et al., 2007). All values have been normalized. Time has also been non-dimensionalized, so that 1 second represents the time that it takes to travel one vehicle length.

$$\begin{array}{lllllll} m & = & 0.0358 & I_z & = & 0.0022 \\ Y_{\dot{r}} & = & -0.00178 & Y_{\dot{v}} & = & -0.03430 \\ Y_v & = & -0.10700 & Y_{\delta s} & = & 0.01241 \\ N_{\dot{r}} & = & -0.00047 & N_{\dot{v}} & = & -0.00178 \\ N_v & = & -0.00769 & N_{\delta s} & = & -0.0047 \\ x_G & = & 0.0014 & Y_r & = & 0.01187 \\ Y_{\delta b} & = & 0.01241 & N_r & = & -0.00390 \\ N_{\delta b} & = & 0.0035. & & & \end{array}$$

Note that the hydrodynamic coefficients given, are not known by the control law, and are assumed to be the actual values that the estimations must converge to.

Simulation results are presented for two cases: one to examine the effectiveness of the proposed control law in the presence of disturbance waves (which result from ocean currents), and another to study the variation of parameters. The initial condition is assumed to be $[y_0, \psi_0] = [0, 30^\circ]$ in all simulations. It is assumed that the disturbance acts as a step wave that is actuated at some time t_1 and is ended at time t_2. Two types of disturbances are examined: one 10% of maximum input value, and the other 20% of maximum input value. In order to examine parameter variations, it is assumed that the variations are sinusoidal with a relatively low frequency (which corresponds to *gradual* variations). We have assumed that the parameter p varies according to

$$p(t) = p + a \sin \omega t .$$

Two cases are considered. For the first case, it is assumed that $\omega = 0.5$ and $a/p = 10\%$, whereas for the second case we consider $\omega = 0.5$ and $a/p = 50\%$. In other words, a 10% variation pertains to

$$p(t) = p(1 + 0.1 \sin 0.5t),$$

and a 50% variation pertains to

$$p(t) = p(1 + 0.5 \sin 0.5t).$$

Note that the control law is not aware of the parameter changes, i.e. the control law is designed for parameters of constant value p, and that the variations are due to unknown environmental effects.

4.1 Results for computed torque control method

The control objective is to track the desired trajectories $[y_d, \psi_d] = [2\sin t, \cos 2t]$. Simulation results are rendered in Table 1.

$\lambda_1 = \lambda_2$	10% Disturbance	20% Disturbance
1	*Unstable*	*Unstable*
2	*Unstable*	*Unstable*
3	*Unstable*	*Unstable*
4	*Stable*, $-60 < \delta_b < 140$	*Unstable*
5	*Stable*, $-50 < \delta_b < 60$	*Unstable*
6	*Stable*, $-60 < \delta_b < 80$	*Unstable*

Table 1. The Required Range of Rudder Deflection For Stability in the Presence of Disturbance, for Different Design Parameters.

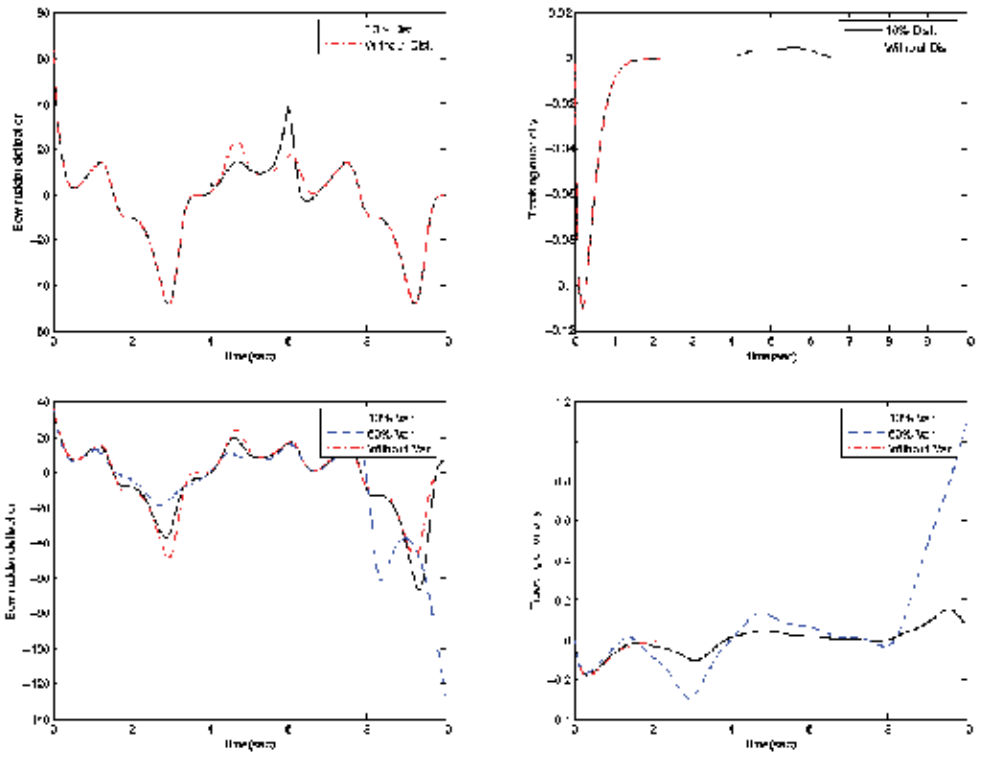


Fig. 2. System's behavior for the computed torque method: (a) δ_b in the presence of disturbance (b) $\tilde{y}$ in the presence of disturbance(c) δ_b in the presence of parameter variations (d) $\tilde{y}$ in the presence of parameter variations .

It can be seen from Table 1, that a 20% disturbance will always lead to instability. Therefore, we only present the simulation results for a 10% disturbance. We also choose $\lambda = 5$, since it requires the least range of rudder deflection, according to Table 1. Fig. 2 shows the rudder deflection δ_b and the tracking error $\tilde{y}$ in the presence of disturbance and parameter variations. Although the tracking error does not converge to zero in the presence of parameter variations, it is still small when $a/p = 10\%$. Tracking error increases with increasing the ratio a/p, and as you can see, a 50% ratio does not yield satisfactory results. Comparing the simulation results of this controller, with the controllers given in the proceeding sections, one can conclude that the controller in this method is sensitive to parameter variations.

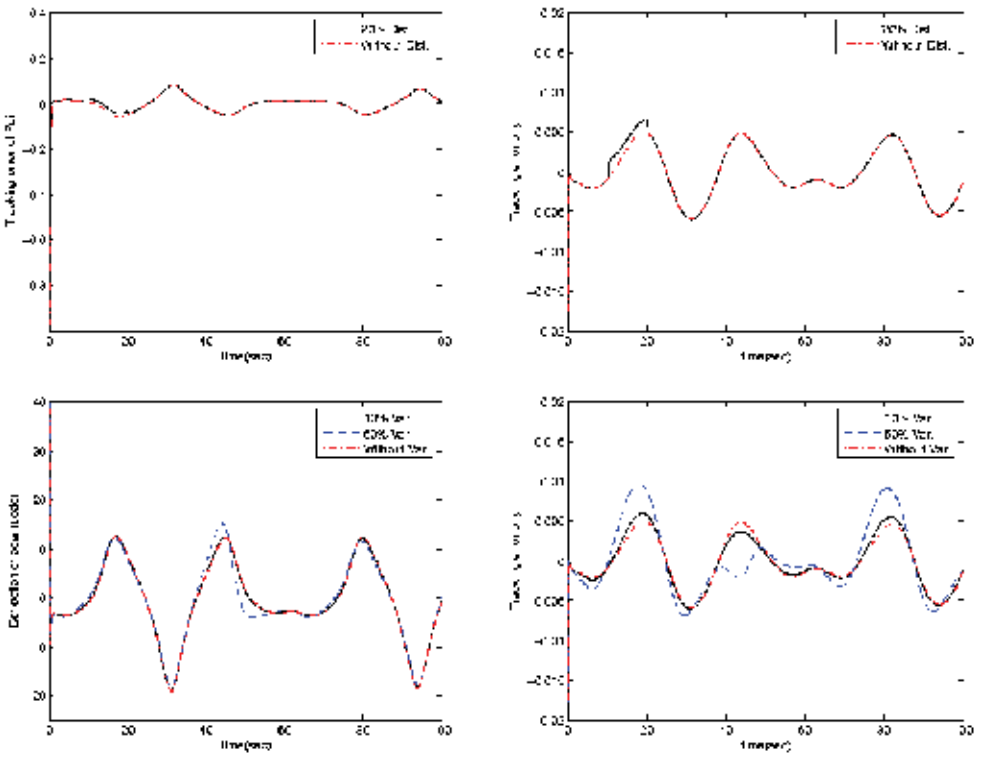


Fig. 3. System's behavior for the adaptive computed torque method: (a) $\tilde{\psi}$ in the presence of disturbance (b) $\tilde{y}$ in the presence of disturbance(c) δ_b in the presence of parameter variations (d) $\tilde{y}$ in the presence of parameter variations .

4.2 Results for adaptive computed torque control method

In this case, it is desired to track the trajectories $[y_d, \psi_d] = [2\sin 0.3t, \cos 0.2t]$. Numerous simulations were performed and it was concluded that a good compromise between control effort and a good response, can be achieved using the following design parameters

$$\phi_1 = \phi_2 = 100$$

$$\gamma_1 = \ldots = \gamma_8 = 0.01$$

$$\lambda_1 = 10, \qquad \lambda_2 = 15$$

Simulation results are shown in Fig. 3. It can be seen that while the computed torque method could not stabilize the system in a 20% disturbance, its adaptive counterpart has led to a successful response. Still more interesting is the system's response to parametric variations: the deviation of tracking error from zero, in the presence of a 50% variation is still small and acceptable.

4.3 Results for suction control

The control objective is to track the desired trajectories $[y_d, \psi_d] = [2\sin t, \sin t]$. The thickness of the boundary layer is taken to be 0.1, with the design parameters $\lambda_1 = \lambda_2 = 5$, and k_1 and k_2 are chosen equal to 10 in the presence of disturbances, and 12 in the presence of parameter variations. The results are shown in Fig. 4. Though simple in design, this method has yield extraordinary results in conquering disturbances and parameter variations.

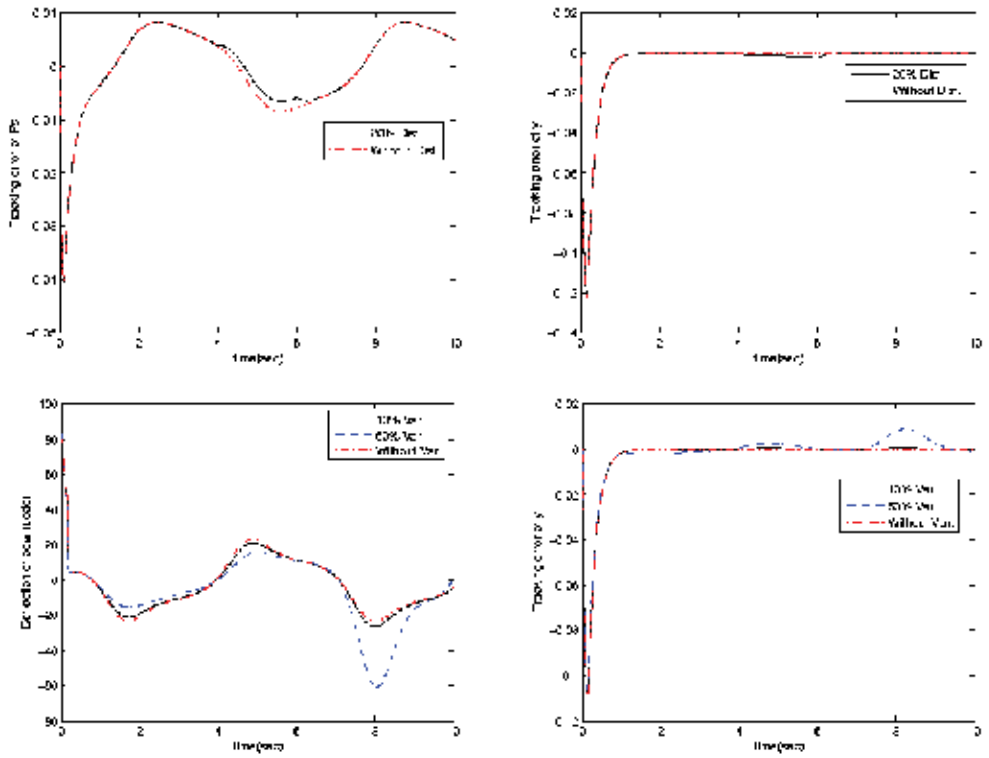


Fig. 4. System's behavior for the suction control method: (a) $\tilde{\psi}$ in the presence of disturbance (b) $\tilde{y}$ in the presence of disturbance(c) δ_b in the presence of parameter variations (d) $\tilde{y}$ in the presence of parameter variations .

4.4 Saturation of rudders

If the control signal generated by the feedback law is larger than possible or permissible for reasons of safety, the actuator will "saturate" at a lower input level. The effect of occasional control saturation is usually not serious: in fact a system which never saturates is very likely overdesigned, having a larger and less efficient actuator than is needed to accomplish even the most demanding tasks. On the other hand, if the control signals produced by the linear control law are so large that the actuator is always saturated, it is not likely that the system behavior will be satisfactory, unless the actuator saturation is explicitly accounted for in an intentionally nonlinear control law design. If such a design is not intended, the gain matrix should be selected to avoid excessively large control signals for the range of states that the control system can encounter during operation (Friedland, 1987).

A conventional value for the saturation of rudders in underwater vehicles is about 30°. (Haghi et. al, 2007) showed that if saturation occurs, the tracking error will not converge to zero, leading to instability of the vehicle. Obviously saturation must be avoided. In attempt to answer *"Why saturation occurs?"*, we overlook the problem definition again. Previously, it was assumed that the vehicle had a constant forward velocity u. The desired trajectory is a sine wave of the form $y_d = a\sin\omega t$, with the amplitude a and the frequency ω. Imagine driving in a road full of sharp turns. Intrinsically, the driver will slow down, to avoid turning over the vehicle. Now if the vehicle's forward speed is constant, then there will be a limit to the frequency of the road turns, that the driver can conquer without smashing his car. Same line of reasoning is made for our underwater vehicle. If the frequency of the desired trajectory ω is too much, the control signal that would be needed to keep the vehicle on the track will increase. If the control signal increases so that saturation occurs, the underwater vehicle will turn over and smash out of the road! Therefore, we conclude that there should be a margin to the maximum value of frequency ω that we can conquer without decreasing the speed, under which saturation will not occur. This value was found for some design parameters λ_1 and λ_2, by making numerous simulations, utilizing the method known as **Bisection method** by numerical analyzers. Simulation results are summarized in Table 2. The value of ω has been assumed to be the same for both y_d and ψ_d.

$\lambda_1 = \lambda_2$	$\omega_{saturation}$
1	1.267578
2	1.181562
3	0.84472656

Table 2. Saturation frequency as a function of design parameters

5. Conclusions and further research

Three nonlinear control methods were proposed for controlling underwater vehicles, and their capabilities to cope with the issues of environmental disturbances and parametric

uncertainties were examined through simulation results. It was shown that the stability achieved by the computed torque control method, is sensitive to parametric uncertainties. Moreover, the maximum amount of disturbance waves that can be conquered by this method was shown to be lower than its adaptive counterpart. The adaptive computed torque control method compensated parameter variations through an adaptation law. As a result, it could manage larger amounts of uncertainties. Finally the suction control method lead to a robust controller, insensitive to uncertainties or disturbances.

The theoretical analysis proposed in this chapter, verified by numerical simulations, has shown that the application of the proposed control laws can lead to a successful design, conquering drastic constraints such as uncertainties and environmental disturbances. The next step in evaluating the efficiency and reliability of these approaches passes necessarily through the practical implementation of such algorithms, and verification of the results by experimental studies.

6. References

Antonelli, G.; Chiaverini, S.; Sarkar, N. & West, M. (1999). Adaptive Control of an Autonomous Underwater Vehicle Experimental Results on ODIN. *Proc. IEEE Int. Symp. Comput. Intell. Robot. Automat.*, pp. 64–69, Nov. 1999, Monterey, CA.

Antonelli, G.; Chiaverini, S.; Sarkar, N. & West, M. (2001). Adaptive Control of an Autonomous Underwater Vehicle Experimental Results on ODIN. *IEEE transactions on Control System Technology*, Vol 9., No. 5, pp. 756-765.

Bejczy, A. (1974). Robot Arm Dynamics and Control. *Jet Propulsion Laboratory Technical Memo 33-669*, February 1974.

Corradini, M.L. & Orlando, G. (1997). A Discrete Adaptive Variable-Structure Controller for MIMO Systems, and Its Application to an Underwater ROV. *IEEE Transactions on Control System Technology*, Vol. 5, pp. 349-359.

Craig, J.J. (1988). *Adaptive Control of Mechanical Manipulators*, Addison-Wesley publishing company.

Craig, J.J. (1989). *Introduction to Robotics: Mechanics and Control*, 2nd edition, Addison-Wesley publishing company.

Cristi, R. & Healey, A.J. (1989). Adaptive Identification and Control of an Autonomous Underwater Vehicle, *Proc. 6th Int. Symp. Unmanned Untethered Submersible Technology*, pp. 563-572.

Cristi, R.; Papoulias, F.A. & Healey, A.J. (1990). Adaptive Sliding Mode Control of Autonomous Underwater Vehicles in the Dive Plane. *IEEE Journal of Oceanic Engineering*, Vol. 15, No. 3, pp. 152-160.

Fossen, T.I. (1994). *Guidance and Control of Ocean Vehicles*, John Wiley & Sons Ltd., New York.

Fossen, T.I. & Sagatun, S.I. (1991a). Adaptive Control of Nonlinear Systems: A Case Study of Underwater Robotic Systems. *Journal of Robotic Systems*, Vol. 8, pp. 393-412.

Fossen, T.I. & Sagatun, S.I. (1991b). Adaptive Control of Nonlinear Underwater Robotic Systems. *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1687-1694, April 1991, Sacramento, CA.

Freeman, R. A. & Kokotovic, P.V. (1996). *Robust Nonlinear Control Design*, Birkhäuser.

Friedland, B. (1987). *Control System Design, An Introduction to State-Space Methods*, McGraw-Hill, New York.

Goheen, K.R. & Jefferys, E.R. (1990). Multivariable Self-tuning Autopilots for Autonomous and Remotely Operated Underwater Vehicles, *IEEE Journal of Oceanic Engineering*, Vol. 15, No. 3, pp.144-151.

Haghi, P.; Naraghi, M. & Sadough-Vanini, S.A. (2007). Adaptive Position and Attitude Tracking of an AUV in the Presence of Ocean Current Disturbances, *Proc. 16th IEEE Int. Conf. Control Applications*, pp. 741-746, 1-3 October 2007, Singapore.

Healey, A.J. & Lienard, D. (1993). Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles. *IEEE Journal of Oceanic Engineering*, Vol. 18, No. 3, pp. 327-339.

Krstic, M.; Kanellakopoulos, I. & Kokotovic, P. (1995). *Nonlinear and Adaptive Control Design*, John Wiley & Sons Ltd.

Leonard, N.E. & Krishnaprasad, P.S. (1994). Motion Control of an Autonomous Underwater Vehicle with an Adaptive Feature, *Proc. IEEE Symposium on Autonomous Underwater Vehicle Technology*, pp. 283-288, Cambridge, Massachusetts.

Marco, D.B. & Healey, A.J. (2001). Command Control and Navigation Experimental Results with the NPS ARIES AUV. *IEEE Journal of Oceanic Engineering*, Vol. 26, pp. 466-476.

Markiewicz, B. (1973). Analysis of the Computed Torque Drive Method and Comparison with Conventional Position Servo for a Computed-Controlled Manipulator. *Jet Propulsion Laboratory Technical Memo 33-601*, March 1973.

Narasimhan, M. & Singh, S.N. (2006). Adaptive Input-Output Feedback Linearizing Yaw Plane Control of BAUV Using Dorsal Fins. *Journal of Oceanic Engineering*, Vol. 33, pp. 1413-1430.

Paul, R.P. (1972). Modeling Trajectory Calculation and Servoing of a Computer Controlled Arm. *Technical Report AIM-177*, Stanford University Artificial Intelligence Laboratory.

Shinner, S.M. (1998). *Advanced Modern Control Systems Theory and Design*, John Wiley & Sons Ltd.

Slotine, J.J.E. & Sastry, S.S. (1983). Tracking Control of Non-linear Systems Using Sliding Surfaces With Application to Robot Manipulators, *Int. J. Control*, Vol. 38, No. 2, pp. 465-492.

Slotine, J.J.E. (1985). The Robust Control of Robot Manipulators, *Int. J. Robotic Research*, Vol. 4, No. 2, pp. 49-64.

Slotine, J.J.E. & Li, W. (1991). *Applied Nonlinear Control*, Prentice-Hall.

SNAME (1950). The Society of Naval Architects and Marine Engineers, Nomenclature for Treating the Motion of a Submerged Body Through a Fluid. *Technical and Research Bulletin,* No. 1-5.

Yuh, J. (1995). *Underwater Robotic Vehicles: Design and Control*, TSI Press.

Zhao, S. & Yuh, J. (2005). Experimental Study on Advanced Underwater Robot Control. *IEEE Transactions on Robotics*, Vol. 21, No. 4, pp. 695-703.

10

Experimental Research on Biorobotic Autonomous Undersea Vehicle

Jianhong Liang, Hongxing Wei, Tianmiao Wang, Li Wen,
Song Wang and Miao Liu
BEIHANG University
China

1. Introduction

On August 2004, the SPC-II robofish, developed by the Robotics Institute in Beijing University of Aeronautics and Astronautics, was used for aiding underwater archaeology in an experiment at the shipwreck site of Donggu gulf in Dongshan Island, Fujian Province. This experiment is not only the combination achievement of many subjects such as bionics, robotics, archaeology, oceanology, and so on, but also the first attempt to apply bionic propulsive mechanism to the real problem.

Fernandez. Benoit, the pioneer of underwater archaeology in France, had said that "the biggest museum in the world is the sea-bottom" [1]. China is an ancient civilization country with a history of more than 5000 years. The voyage technique was used to be vigorous and business on the sea was very frequent, so we can imagine that there are abundant of cultural relic imbedded in the 3,000,000 sq.km sea area. In 1987, the laboratory of underwater archaeology was founded by the historical museum of China and then the first team of underwater archaeology was founded in China in 1990[1].

Now the underwater archaeology procedure can be divided into three phases. The first is large-scope searching. Usually the equipments such as sonar are carried by archaeological ship to find the doubtful site. Sometimes this work can be replaced by the information offered by fisherman. The second is the site validation. The equipments such as lighting and vidicon were carried to the sea-bottom to do close observation, and the pictures or sample objects are sent back to do more analyses. The third is salvage and excavation. It need long time to do underwater work such as divide region, clear bed load, recover site, salvage cultural relic and so on.

In the above three phases, the last two are more pivotal and need to be performed on the sea-bottom. Normally they are completed by the archaeological team members. It can be seen that underwater work plays an important role in the task of underwater archaeology. Because of the restriction of human body physiology (narcosis of nitrogen), generally people can only dive to 60m to70m in depth. And meanwhile people are weightless in the water, movement inconvenience, overdraft of physical strength, short duration, and needing long time to be decompressed, all make the work inefficient[2]. The early underwater archaeology was difficult, and the deepwater archaeology was almost impossible in a manner.

The development of underwater robotic technology brought new chance to underwater archaeology. In 1989, Doctor Robert Ballard in U.S. used a ROV named 'Jason' to investigate an old shipwreck site in Med and 'Jason' successfully screened the shipwrecks photograph from the sea-bottom 800 meters under water. With the help of underwater robot, archaeological team can explore anywhere without the danger of life. In recent twenty years, China has also paid more attention to the development of underwater robotic technology. With the development of underwater archaeology and the high need of equipments, underwater robots are gradually applying to underwater archaeology. In 2001, in the archaeological research of underwater old buildings in Fuxian lake of Yunnan province, ShenYang Institute of Automation used ROV named 'goldfish No.1' and AUV named 'CR-02' for assistant diving to about 70 meters under water, to send real-time images of lake-bottom back to archaeologist for analysis. Although the underwater robot (used for observation) can do the second phase work of archaeological team, it is still difficult to replace people to work in the third phase, which is an important research topic of underwater robot in the future (used for busywork).

SPC-II is an underwater robot using bionic propulsive mechanism. Compared with thruster of traditional screw propeller, it has some traits such as high efficiency, good maneuver-ability, low noise, manipulation's facility, small dimension and weight, etc. Concern to the trait of SPC-II, this paper discussed the feasibility of underwater work, and a scheme that SPC-II can help to complete the second phase work of underwater archaeology was proposed, which was validated by experiment. In the end the improvement method was summarized.

1.1 Requirement of underwater archaeology to UUV

Because of the particularity of the environment and objects of underwater archaeology, there are some special demands for the robot and the equipment, which are described as follows:

The disturbance and destruction to the environment should be as little as possible. It is hard to avoid shuttling in the site back and forth when the underwater robot is working and the result would be serious if the cultural relic was shattered. The silt aroused by disturbance can also affect the collecting of image information. The robofish, adopted the bionic propulsion, has its own advantages compared to the traditional screw propeller. In addition, the electric cable of ROV is a potential trouble, and the crawling robot and daggling robot do not adapt to the archaeology.

Convenient equipments should be prepared for the task. The exploration apparatus and other equipments must be equipped for the underwater robot for archaeology, and they must not do harm to the cultural relic. Generally the underwater TV and sonars are used. But the working depth of underwater TV is limited, and it must work with headlamp when the depth is more than 15 meters. Sonar is the dominant equipment for underwater exploration[3]. Figure1 (a) and (b) are the images returned from underwater TV and side-scan sonar, both of which have their advantages and disadvantages respectively. The sight depth of the underwater TV is short and the eyeshot is small but the image is legible; and the image of the side-scan sonar is blurry but the sight depth is long and the eyeshot is wide. If the underwater robot was applied to the third phase of underwater archaeology, the equipments, such as machinery arm, manipulator and special tools must be equipped with it.

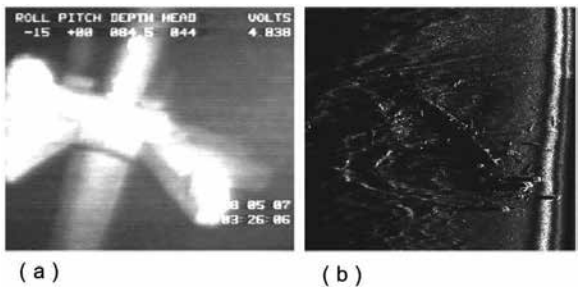


Fig. 1. Images from underwater archaeology

The Dynamic Positioning should be accurate. How to guide the underwater robot to the known target (accuracy) and how to hang around the target by propeller control in order to perform further step task after finding out target (repeatability), all of which need the support of underwater acoustical position system with high accuracy.

High maneuverability. It contains the maneuverability in horizontal (turning radius and turning rate) and vertical (rising and submergence). Usually the shipwreck site is rough and complex, so the robot maneuverability can guarantee the success of the archaeology task.

The robot should have local intelligence. It could track the sea-bottom landform and avoid obstacles (independently) autonomously. Generally, the stable depth of the underwater robot is relative to the sea level. But the archaeology exploration task needs to follow the contour line relative to the sea-bottom, which involves tracking of sea-bottom landform. In addition, for the sake of safety, the robot ought to have the capability to avoid obstacles independently.

The robot should have small size, light weight and low cost, which are considered from application and popularization. Robot with large size and heavy weight could bring trouble when putting and retrieving. In order to enhance the maneuverability, the underwater robot with traditional screw propeller usually need more than one thruster to produce the propulsion of vector, which makes not only the large size structure, complex control system, but also the high cost. Nevertheless the robofish has certain advantage because it produces the propulsion of vector directly by tail fin and is easier to integrate into a small size.

The control should be convenient and flexible. Generally, the operators do not know about the situation of sea-bottom when the underwater robot is carrying out the archaeology task, which are navigated by the images and information returned from underwater robot. As a result it needs the underwater robots have good maneuverability, which is also for security.

Moreover, there are many common problems such as the real-time exchanging of information between the mother ship and underwater robot (underwater communication), the dynamic hermeticzation with high compression resistance, the ability of resisting ocean current and so on.

1.2 Feasibility of SPC-II to perform underwater task

1) Introduction of robofish SPC-II

The bionic propulsive underwater vehicle was researched in the Robotic Institute from 1999. "S.P.C.", which was brought forward on the basis of our many years' research and experience, is a design principle of the robot, which means that stability is superior. The series of SPC-Irobofishes, designed in "S.P.C." mode have attained good effect. SPC-II is the revised version of SPC-I, which was developed for some engineering application, but still in the phase of theory demonstrating and validating.

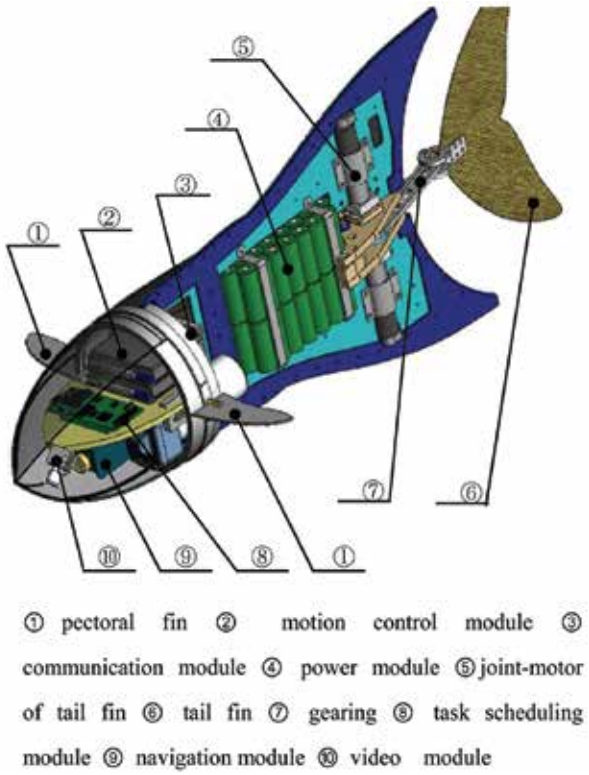


Fig. 2. General layout of SPC-II

SPC-II adopts streamline shape and FRP (Fiberglass-Reinforced Plastics) material. It is mainly consist of eight parts: console, communication module, mechanical reality, power module, task scheduling module, motion control module, navigation module and video module. The robofish can perform five basic actions of going ahead, turning left, turning right, rising and submergence, and it can achieve the function of course keeping, submergence maneuverability, swimming in certain depth, GPS navigation, video capture. Figure 2 is the general layout of SPC-II.

Figure 3 shows the system framework and data flowing.

In order to communicate with SPC-II conveniently, we develop the console software based on PC. The main functions are as follows:

System Monitoring. It can check the work state, water leak state and system communication state periodically, and raise the alarm when there is abnormity.

Setting Parameters. Regulating the swimming parameters of robofish, and setting sea route and constituting the task.

Dynamic Manipulation. Real-time Controlling heading, velocity, ascending or submergence of the robofish, and showing the information returned from sensor in graphics mode synchronously.

Data recording. Record the information of swimming track, image, system state and so on.

Figure 4 is the main interface of the console software, which can be divided into several areas according to function. The main manipulation is related with function keys of the keyboard and the operation is simple, speedy and agile.

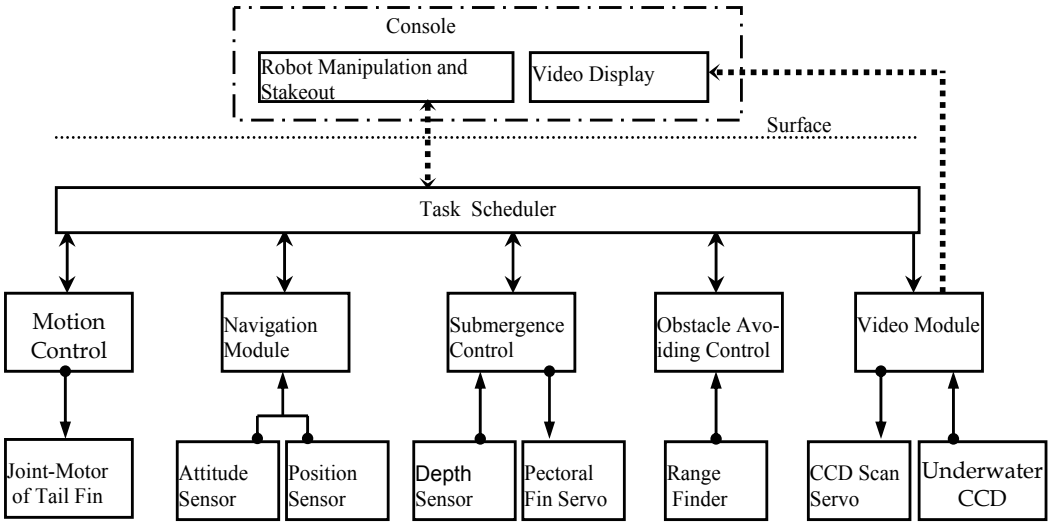


Fig. 3. System framework

Fig. 4. Main interface of console software

2) feasibility of executing underwater task

The robofish SPC-II adopt two servo-joints that are controlled separately. It is benefited to regulate the tail locomotion parameters accurately and obtain better flicking waveform. SPC-II reaches a steady speed of 1.4m/s at frequency of 2.5Hz and its proportion of velocity to body length is more than 1 L/s.

Flexible multi-joint fish robot has been proved to have perfect maneuvering performance. For example, 4-joint VCUUV turns at 75 °/s with one body length in comparison with 7

body lengths of a conventional UUV [4]. SPC-II with rigid body is reduced to only 2 joints, but it still can easily turns with 1 body length turning radius. Usually the small turning radius can obviously improve the efficiency of UUV's employment. Moreover, SPC-IIuses 2 elevators fixed to the part of pectoral fin to dive.

SPC-II carries power battery itself and can communicate with console through cable or wireless style, which makes the transformation between ROV(Remotely Operated Vehicle) and AUV(Autonomous Underwater Vehicle) very simple. Compared with the conventional ROV, SPC-II has the character of higher sailing speed; and compared with the conventional AUV it also has the merit of small size, good maneuverability, and flexible manipulation. At present SPC-II was equipped with a kind of nickel-hydrogen pile (48V, 500wh), with which SPC-II could cruise about 3.5 hours (at 2 kts.). If the lithium pile with the same volume (48V, 1000 wh) is used instead, the cruising time could reach 7 hours.

The comparison between SPC-IIand some other UUVs is described as Table1. Besides, SPC-II still reserves a loading room of 9kg for some special task, which means it still can be added relevant equipment according to real requirement. At a word, SPC-II can perform simple underwater task after proper refitting and has its own advantages and specialties in efficiency and quality.

Name/ Manufacturer	**CR-01(AUV) / ShenYang Institute of Automation**	**VCUUV/ Lab of C.S.Draper**	**SPC-Ⅱ /Robot Research institute of BUAA**
Length/ Displacement	4.374m /1305kg	2.4m /136kg	1.21m /40kg
Battery Duration	10h	3h	3.5h
Dive Depth	6000m	10m	5m
Maximum Velocity	2 kts	2.4 kts	2.8 kts
Turning Rate / Radius	____	75°/s /1 L	70°/s /1 L
Propeller/ Displacement	____	32%	10%

Table 1. Comparison between some UUVs

1.3 Underwater archaeology experiment

At present, SPC-II can elementarily fulfill the need of underwater archaeology and it also has some advantages in environment disturbance and horizontal maneuverability. But the ascending and submergence of SPC-II are implemented by changing the pectoral fin's incidence angle, so it can dive only in the process of swimming (submergence velocity is the minimum straightforward velocity of submergence). In addition, it can not swim back off and hasn't the underwater position system. All of which bring more difficulty to underwater pointing task. Moreover, SPC-II is only a robot prototype with simple equipments; it can only help the archaeological team to complete the second phase task.

Donggu gulf site of shipwrecks in Dongshan city, Fujian Province is in a gulf, and the offing is much calm, but the water quality is much turbid (especially when is ebbing). The position of site has been known that is in less than 5 meters. The benthal hypsography likes gently

brae, and the diffused china, iron weapon and ships were enchased or buried in silt. The purpose of our attempt is to let the robotfish dive to the site and capture and feedback the valuable underwater image information by the vidicon. The implementing methods are described in detail as follows.

1) Experiment scheme of robotfish aided underwater archaeology

The communication chain between the SPC-II and the console was properly modulated. As a result, the communication with cable and the wireless communication are both adopted (buoy and radio-communication as shows in figure. 5). This method not only saves the expensive underwater communication sonar, but also avoids the encumbrance result of too long cable.

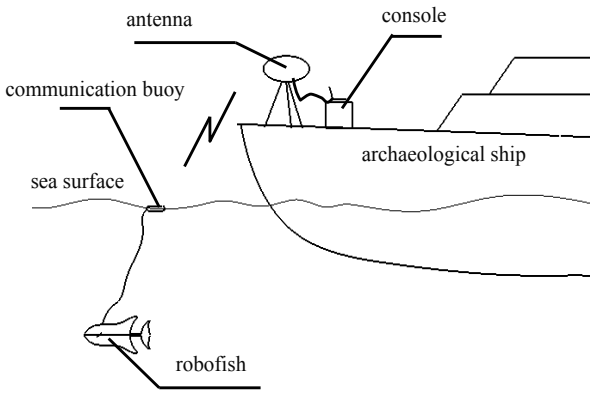
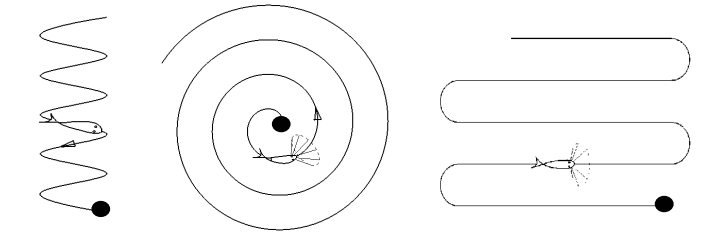


Fig. 5. Sketch map of system communications

The method of hover submergence to explore the object was developed, which would resolve the problem that SPC-II could not hang when swimming and could not ascend and dive vertically. Because the turning radius of robotfish is small and the spiral velocity is slow, and the robotfish can keep swimming in stable depth, it can screen the object distinctly from any directions. The process can be divided into two steps: 1) arriving aim area; 2) scanning and searching unidentified objects. It can be decomposed into several sequences of robot actions or control operations. First the robotfish swam on the sea level at high speed to reach the above of the aim area with GPS navigation; then it spirally dived to appointed depth in short turning radius (Figure 6 (a)). And now the robot had reached the aim area. Two measures were implemented for scanning and searching unidentified objects. The first is that the robotfish hovered in stable depth at diving velocity and gradually augmented the hover radius along the involutes (Figure 6 (b)). Moreover the second is that it swam out and home in stable depth at diving velocity and gradually made lateral move (Figure 6 (c)).

The robofish started the video module to capture and the console started to monitor synchronously and recorded the video information during the process mentioned above. Once discovering the questionable objects, the robot hovered at the place and modulated the submergence depth in order to receive more clearly and abundant image information. But in practice, the sea-bottom gradient is changing and SPC-II does not have the ability of landform tracking, so the searching effect in stable depth (3 meters) was not perfect, and it usually appeared the phenomenon that the level of sea-bottom had overstepped the

camera's eyeshot. Consequently, the method, that SPC-II swam in the ordered height little higher than sea bottom and explored the sea-bottom all the time, was proposed. It had been proved to be much fit for the even landform.



(a)hovering dive (b) hovering scan (c) traversing scan

Fig. 6. Methods of robofish aided archaeology

2) Experimental result

The practice has proved that the strategy mentioned above is feasible. The robofish can dive to target area to search and can cover it on the whole, at the same time, the images of target are returned . But the efficiency is low and the eyeshot of single image is too narrow, which could not form the panorama vision yet. Comparison of the two measures for searching questionable objects has shown that the efficiency of the first is higher than the second in theory, but its control is complex, and the robofish is hard to swim according to the scheduled track, which could result in omitting repeating scanning. But the operating of the second is easier compared to the first. Table 2 gives the result of several methods of searching for an square area of about 200 square meters.

Search Method		Time Consumed	Number of dubious site
Hovering Scan	Keeping Settled Depth	773s	6
	Keeping Close to Sea-bottom	955s	10
Traversing Scan	Keeping Settled Depth	843s	7
	Keeping Close to Sea-bottom	1102s	15

Table 2. Comparison between the results of different methods

It is obvious that although the helical scan wasted less time than out and home scan, but the information of the image returned was little, which was the result of area omitting, and which was almost unavoidable in the condition of manual operating. Moreover, exploring the bottom could find more questionable objects than searching in certain depth, but the cost

was that it raised the benthal deposit and influenced the quality of images. Once the robot was equipped with underwater positioning system and side sonar, it would achieve the autonomous swimming of the foregone trace and the tracking of benthal landform, which would improve the exploration efficiency and quality.
Figure 7 shows a questionable picture returned from SPC-II. Since the seawater is turbid and the target is too close to the lens, its field is very blurry.

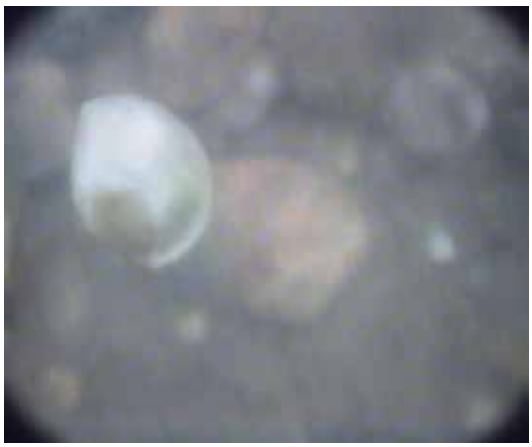


Fig. 7. One returned image from robofish

1.4 Conclusion

Underwater archaeology experiments have validated the performance of SPC-II. Meanwhile, some problems have been exposed, which referred to Dynamic Positioning, maneuverability (vertical), communication, configuration of functional devices and diving depth, etc. All these problems supply references to future researchers. To serve the underwater archaeology better, and build basis for the future complicated applications, the robofish SPC-II should be improved properly. The details are described as follows.

1. Add an underwater positioning system. Influenced by ocean current and stormy waves, the existing dead reckoning error is unacceptable. It makes the acoustic positioning system necessary. This system is also strongly required for dynamic positioning.
2. Improve the maneuverability of vertical plane. One of the diving methods is changing its own weight, but it is difficult to respond dynamically; and the other is changing the incidence angle of pectoral fins, but it requires a certain speed. In order to dive faster and vertically, it is considered to replace the pectoral fins with two vertical screw propellers.
3. Add high-performance functional devices such as high accuracy and wide-eyeshot CCD cameras, side-scan sonar, robot arms, manipulators, etc.
4. The diving depth. The target depth is set to 100 meters. Most of sites under water are within this range.
5. Optimize local intelligence. This needs to upgrade the corresponding sensors and improve algorithms.

2. Propulsive and maneuvering performance of two joints biorobotic autonomous undersea vehicle SPC-III

2.1 Introduction

Intensive study has been carried out in respect of the outstanding swimming skill of some marine animals such as dolphin and tuna in the recent 20 years. Focusing on their prime propulsion device, i.e. their caudal fin, the researchers has preliminarily found out the locomotion of body-caudal fin and hydrodynamic principle [5] [6]. Parameter range has even been worked out to achieve optimum kinematics of caudal fin flipping by conducting modeling, simulation calculation and model experiment in the pool [7][8][9][10]. The results of these researches have facilitated the birth of the Prototype of mission-scale biorobotics autonomous undersea vehicle. Low speed maneuverability, lower noise and high efficiency have become hot issues [11]. Anderson et al. developed the Vorticity Control Unmanned Undersea Vehicle. The 2.4m –long VCUUV follows the morphology and kinematics of a yellow fin tuna and adopt hydraulic-power 4 joints tail drive structure. It attained a speed of 1.2 m/s, and a yaw rate of 75°/s. [12-13]. Nakashima et al. developed a two joints dolphin robot, which is 1.75 m long. The robot attained a propulsive speed of 1.2 m/s and a propulsive efficiency of 0.35. [14-16]

Yet all of these achievements are not convincing enough to talk the chief designer of autonomous undersea vehicle into adopting bionic propulsion technology in his AUV project in the near future. Tangible proofs need to be put forward to show that biorobotics propulsion has remarkable advantages in some respects. At the same time, other factors also require consideration such as payloads, endurance and stability, so as not to lose too much performance of the vehicle.

Biorobotic autonomous undersea vehicle SPC-III, developed by Robotics institute, BeiHang University, is a prototype of mission-scale autonomous submersible vehicle. According to its displacement tonnage, it can be classed as portable UUV [17]. The purpose is to develop an available biorobotics propulsion device on the basis of the understanding of the mechanism of caudal fin in itinerant state as well as the current mature engineering technology. The said biorobotics propulsion device should not only improve maneuverability and reduce noise of the conventional propulsion UUV but also have acceptable speed, propulsion efficiency or endurance. Researches on SPC series biorobotics unmanned undersea vehicle started from 2001. SPC-I is a prototype used to study what impact navigation stability and yaw may have on tail fin propulsion [18]. SPC-II is a prototype used to study hovering and turning maneuverability. It achieved a yaw rate of 30o/s and a turning diameter about its body length. In 2004 it was applied in an attempt to the mission of Underwater Archaeology in Dongshan Island of Fujian undertaken by the Underwater Archaeological Team of the National Museum of China [19-20]. SPC-III has the same torpedo shape with the conventional UUV. Its two joints caudal fin thruster can be easily replaced with screw propeller. Since the hydrodynamic shape of the two are almost the same, preliminary results of the performance of various propellers has been achieved, which are used to evaluate the feasibility of applying the designed biorobotics unmanned undersea vehicle to probe commission.

The rest parts of this article are arranged as: Part 2 is a detailed description of Biorobotic autonomous undersea vehicle SPC-III, especially the realization of two joints caudal fin thruster; Part 3 presents the comparison result between propulsion power, turning radius as well as rate and screw propeller; Part 4 is a description of the result of the long-distance

probe test carrying Water Quality Multiprobes; Part 5 discusses the performance and problems of SPC-III; Part 6 is the conclusion of this paper and the prospect for the future.

2.2 Description of the experimental system

1) Description of SPC-III System

SPC-III is designed to be a mission-scale vehicle; therefore its hull should have the ability to bear the pressure of water. The forward and middle sections of the hull are constructed with carbon fiber and aluminum alloy framework. The caudal section is an installed caudal fin thruster. Due to its complex shape, it is molded in the die using engineering plastics. SPC series biorobotics vehicle has two important characteristics: (1) the caudal fin thruster does not use flexible structure but expose Drive link assembly which has small volume and low weight in water. Thus conventional pressure hull and seal style can be adopted without considering the additional power consumed by flexible material for swinging deformation. (2)caudal fin is driven by two actuating DC motors, which is a comparatively mature robotics technology realizing accurate motion of caudal fin. Compared with other modes, DC actuating motors has higher electro-mechanical conversion efficiency.

As is shown in Figure 1, the forward half of SPC-III is comprised of equipment and the payload capsule, with the IFLY40 Autopilot used to control Small UAVs installed at the front. IFLY40 was successfully applied to the 24th Chinese National Antarctic Research Expedition in December 2007, finishing the task of sea ice investigation [21]. It can provide the data of posture, course angle, position and altitude and has five PID control devices. The only difference is that the barometric sensor is replaced with the water pressure sensor, which provides depth control of 0 ~ 50m. Below the Autopilot is an instrument for AD transfer and recording. At the back is the movement controller of caudal fin thruster, which will be introduced in details in the following part of this paper. The middle section is the dynamic cabin, which is composed of dynamic units formed by 28 pieces of 84-watt Li-poly batteries and providing permanent endurance for SPC-III. On the top of the dorsal fin is GPS antenna of the Autopilot.

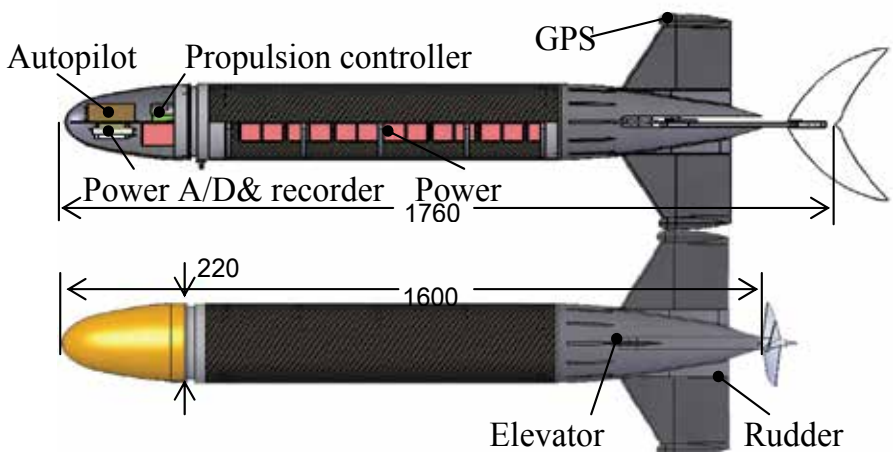


Fig. 8. SPC-III biorobotics AUV and the Comparison AUV with a screw propeller

2) Caudal fin thruster and its control law

As is shown in Figure 2, caudal fin which generates thrust forces is installed at the end of the Drive link assembly. The caudal fin is made of 1mm-thick carbon fiber and imitates the shape of caudal fin of tuna, but with a lower aspect ratio. Dimension data is shown in Table 1. Link 1 and Link 2 are respectively driven by a Maxon RE40 24V motors trough reducers with reduction ratio of 30. Let flapping amplitude of the caudal fin be A, attack angle of relative stream be α, and θ_1, θ_2 be the respective output angle of the two motors through the reducer. And then the relationship can be expressed as:

$$\begin{cases} A = L\sin\theta_1 \\ \alpha = \theta_2 \end{cases} \tag{1}$$

In itinerant state, the performance effectiveness of caudal fin is affected by the following parameters [13]. They are (1) Dimensionless flapping amplitude, which is defined as $H=A_0/c_0$, where A_0 is the peak of flapping. (2) attack angle amplitude of caudal fin:α_0. (3) phase difference of the former two: Φ. (4) Strouhal number, which is defined as $St.=fA_0/V$, where. V is the speed of inflow. Motion law of caudal fin of caudal fin propulsion marine animals like dolphin and tuna can be expressed as:

$$\begin{cases} A = 1/2A_0\cos(2\pi t) \\ \alpha = \alpha_0\cos(2\pi t - \phi) \end{cases} \tag{2}$$

Thus it is clear that accurate adjustment of the former three factors can be realized only by adjusting motion law of output$\theta_1\theta_2$ of motors. As the control system of caudal fin thruster, a 2-Axis motion coordinator is used generate the above motion law. As for the optimum range of the four parameters, previous work has derived preliminary conclusions [7-9][22]. Part of the results obtained by the self-propelled SPC-III in open sea will be shown in the experiments introduced in the following part of this paper.

As it is difficult to install two driving units in the narrow cone space of the caudal section, a very thin spur gear reducer is customized. At the same time, motors—reducer-sealing assembly are put in staggered arrangement to complete the assembly of caudal fin thruster. Thus it is impossible to add more torque and speed sensors.

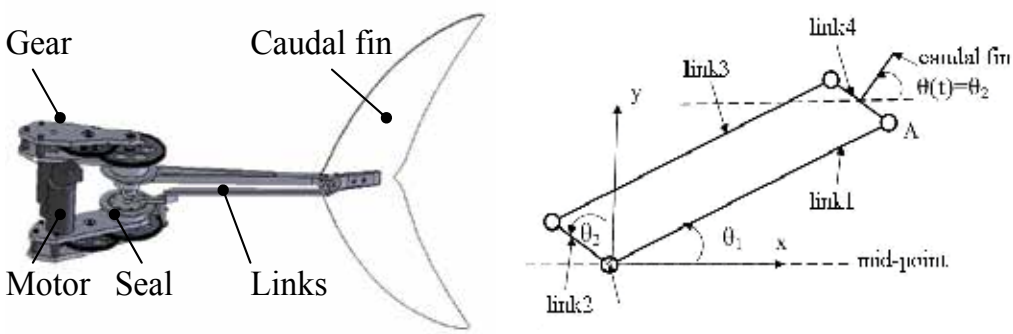


Fig. 9. The mechanical sketch of caudal fin thruster

3) Propeller comparing AUV

The building of a AUV experimental platform used for comparison is actually to directly replace caudal fin thruster with screw propeller on SPC-III, while dynamic units, motion coordinator and amplifier remain the same. Thus difference factors are reduced. The selection and production of the propeller have got the help of China Ship Scientific Research Center (CSSRC). Open water efficiencyη_0 of the propeller in five sections is predicted to be 0.67 [23]. The structure of screw propeller is shown in Figure 3. Parameter comparison between it and caudal fin thruster is shown in Table 1.

Fig. 10. Structure diagram of the screw propeller

Caudal fin propeller		Screw propeller	
Area of caudal fin: S (mm²)	25287	Diameter: D (mm)	240
Maximum chord length: c_0 (mm)	120	Number of blades: Z	3
Average chord length: c (mm)	70	Expanded area ratio: A_E/A_0	0.36
Lead edge sweepback (deg.)	47	pitch ratio (P/D): $_{0.7R}$	0.837
Airfoil	1mm flat plate	Airfoil	NACA66mod a=0.8
Length of Link1,3 (mm)	280	Hub radial ratio: d_h/D	0.18
Length of Link2,4 (mm)	28	Rotation direction	Right
Driving motors	RE 40 (150w) ×2	Driving motors	RE 40 (150w)×1
Reducer	i=30 (grade 2 spur gear)	Reducer	i=5 (grade 1 planetary gear)
Weight (kg.)	3.3	Weight (kg.)	2.1

Table 3. Comparison between caudal fin thruster and screw propeller

2.3 Experiment on propulsion and maneuverability

1) Measurement system

The test system of the complete SPC-III is shown in Figure 4. IFLY40 Autopilot provides it with perfect telemetry function. This paper is focused on power of the propeller, as well as speed, maneuverability of the vehicle. All collected data are added into the protocol through Autopilot before they are sent to the ground station software. The transmission speed is adjustable within the range of 1~10 frames.

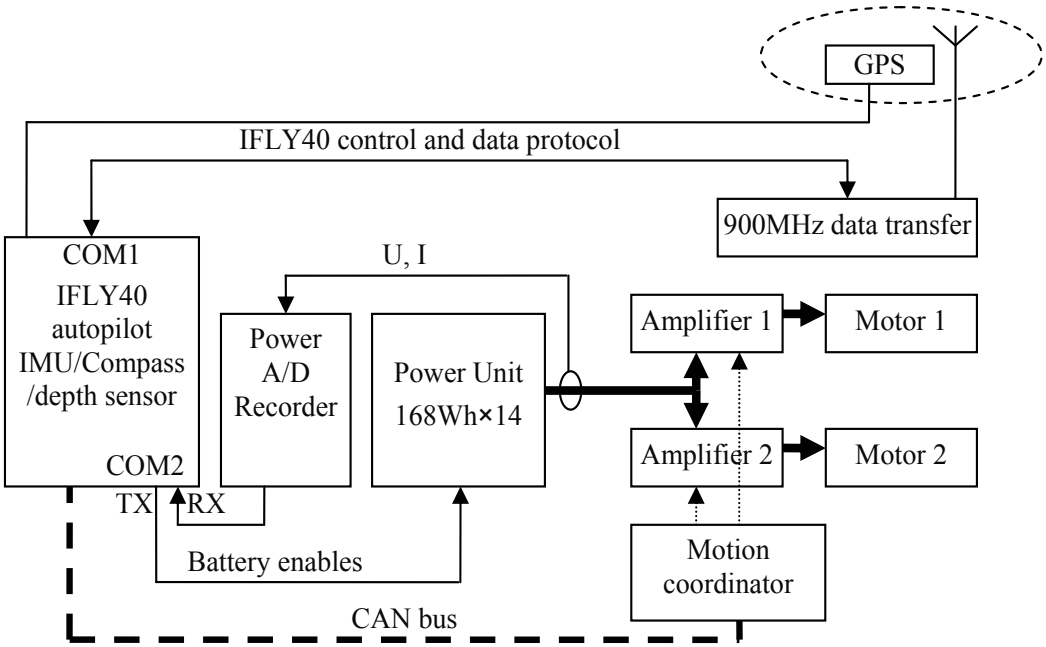


Fig. 11. SPC-III AUV electrical system

- **Vehicle velocity measurement.** IFLY40 Autopilot possesses various navigating modes such as UAV and RPV, which provides advantage for the vehicle to keep course or route for a long time to have tests. Therefore, this paper directly takes the longitude-latitude co-ordinate sent back by Autopilot as the original data to calculate distance and speed. Positioning of the GPS has accuracy as high as 2.5m(CEP). Comparatively accurate average speed can be achieved if distance long enough is used for calculation. In this paper, data of 200 seconds are used for the calculation of distance and speed.
- **Power measurement**. A 16-bit 8-channel A/D embedded computer is adopted for the measurement of power. Sampling frequency is 100Hz. Integrating Calculation over power is carried out every second. Through Autopilot, power obtained is sent to Ground Control System (GCS) software, which records the data frame. Under the modes of Vehicle velocity measurement and power measurement, transmission speed of Autopilot is one frame per second. Similar to Vehicle velocity measurement, average value is also got by calculating data of 200 seconds under power measurement. Nodes of current and voltage measurement are set on the circuit through which the batteries supply power to the amplifier; therefore the power measured is all the power that consumed by the propeller.
- **Yaw rate measurement.** Under the mode of maneuverability measurement, the speed of Autopilot increases to 10 frames per second. IFLY40 adopts data fusion algorithm to synthesize outputs of IMU and electronic compass into a course angle that is hard to be disturbed and can cause drift.

2) The comparison of propulsion performance

From the 9th to the 13th of October, 2006, Comparative experiment on SPC-III and its propeller was carried out on the coast of Qinhuangdao which is located in the west coast of the Bohai Sea. As is shown in Figure 5, the experiment was carried in calm offshore waters

at ebb time every day. Firstly, static power and zero-load power of the propeller were measured. The results show that the power of caudal fin and screw propeller at static state are respectively 7 Watt and 3.5 Watt, correspondent to the static power of two amplifiers and one amplifier. Zero-load power refers to the motion power of the propulsion system except caudal fin and screw, including all transmission loss. At this stage, because of more complicated mechanical structures, the power of caudal fin thruster is higher than that of screw propeller. The former is 25 watt at 2.5Hz, while the latter is 18watt at the speed of 10 turnings per second.

Fig. 12. SPC-III biorobotics AUV（the right one） and the Comparison AUV（the left one） on the beach

Measurement in water. Firstly, the posture of the vehicle was adjusted to level. To ensure that GPS could receive stable signals and that the wireless data transfer device could work in good condition, draft was adjusted until GPS was 0.2m above water. To avoid influence from the control system, instead of starting the course control function of the Autopilot, the neutral position of caudal fin flapping or the deviation angle of rudder were adjusted manually to keep the vehicle in a linear trajectory. Note that the rudder is disabled when caudal fin thruster is used.

The measurement of power and Vehicle velocity was carried out continuously. After Autopilot was set with new frequency, rotation speed and flapping parameters through Ground Control Software GCS300, the vehicle was meticulously adjusted to keep it in a linear trajectory. This stable status was kept for a few minutes to allow the GPS time recorded. Relevant measurement data was searched according to the GPS time for later data processing. As is shown in Figure 6, 7,8, the final test results indicate that at 2.5Hz, H=1.5, caudal fin thruster achieves maximum speed 1.36m/s. At that moment, the total power of the propeller is 161watt; screw propeller achieves maximum speed 1.4m/s at the speed of 10 turnings/sec. At that moment, the total power of the screw propeller is 165watt. The pitch of the vehicle became unstable when the flapping frequency or rotation speed was increased to increase Vehicle velocity. Another reason may be that resistance at water surface is higher than that underwater, so that vehicle is prone to pitch and oscillation or it may submerge into the water, so that GPS can not work.

The Principle of motion law adjustment for caudal fin flapping is to generate comparatively greater thrust forces and speed for the comparison with screw propeller. Because of deformation of the carbon fiber caudal fin, there is some inconsistence between the actually

applied motion law and the ideal value obtained through earlier work. Deformation of the caudal fin may be explained as the increasing of attack angle and the lag of phase. Finally, the values adopted in the experiment are: α=10°, Φ=45°. At that moment greater propulsion speed is achieved. Measurement was carried out respectively when H=0.75, H=1, H=1.5, and it is discovered that when H=1.5, the output power is higher at similar frequency, and power consumption is lower at similar speed.

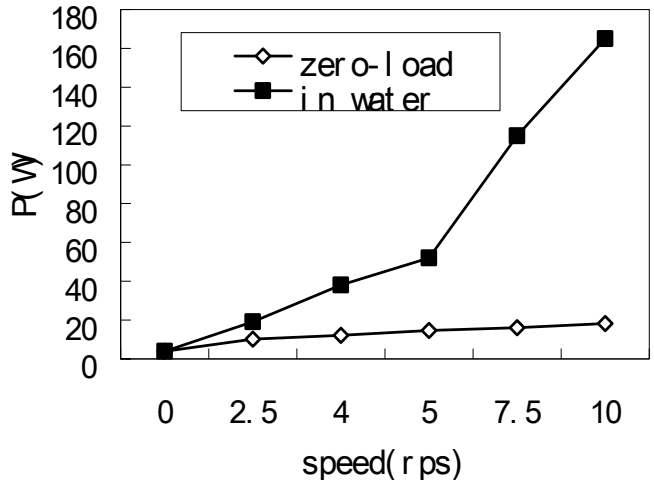


Fig. 13. Power of screw propeller in Linear Motion, „zero-load" means the screw was not installed.

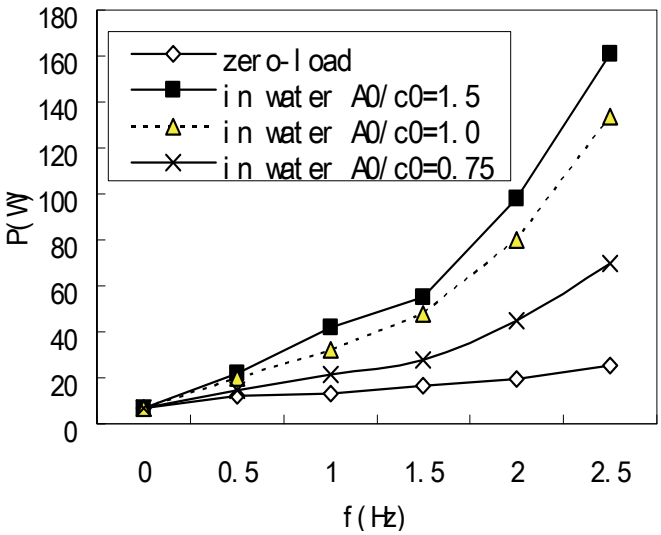


Fig. 14. Power of caudal fin thruster in Linear Motion, „zero-load" means the screw was not installed.

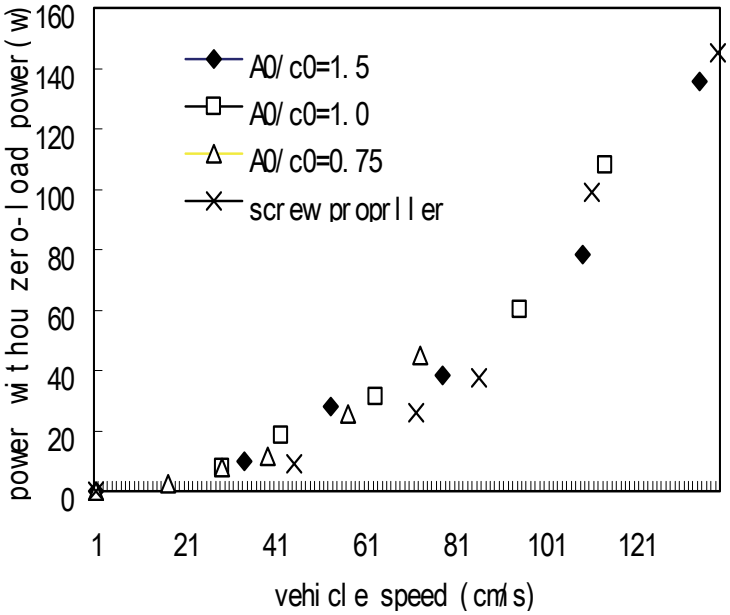


Fig. 15. Power consumption (without zero-load power) of caudal fin thruster and screw propeller

3) Measurement results of maneuverability performance

Possible commissions of portable UUV include probe of port and coast, as well as identification and destroy of torpedo [17]. To perform these commissions, AUV often needs to be close to the object in a small distance and at the same time avoids colliding. At that moment low speed maneuverability is particularly important. For example, AUV is often constrained in a narrow space when it is in the state of autonomous navigation. At that moment AUV needs to turn in situ round itself to go back to open sea, which is a maneuverability often used by ROV but difficult for AUV whose advantage lies in its cruising.

The VCUUV achieved 1.2m/sec and turn rates up to 75°/sec[12]. With flexible caudal hull and four joints caudal fin driven by hydrodynamics power, VCUUV possesses excellent maneuverability. It achieves a turning diameter of two body lengths(BL). Though the hull of SPC-III is completely rigid, and it only has two joints, its special caudal structure enables caudal fin to realize a deflection angle of 0~90°. 90°deflection angle can be used for emergency braking.

Figure 9, 10 shows the circular trajectory of SPC-III and propellers comparison AUV under maneuverability measurements. The trajectory was drawn at GCS300 ground station software using GPS Coordinate data recorded by Autopilot. Note that the calibration of the map scale is 5m. SPC-III adopts a flapping frequency of 2Hz, with a rotation speed of propeller at 7.5r/sec, and correspondent linear speed being about 1.1m. Yet the speed decreases remarkably as the turning radius decreases in turning state. At 45°deflection angle, caudal fin thruster achieves a turning diameter of 2.5BL, while screw propeller which uses rudder achieves a turning diameter of 5BL. Caudal fin thruster achieves minimum

turning diameter 2BL at 60°deflection angle. Figure 11 illustrate the results of turning speed measurement, including two kinds of data. The first data are obtained through calculation according to the time the vehicle took to finish circular route; the second data are obtained according to compass data. The two kinds of data are almost the same. At similar deflection angle, the yaw rate of Propellers AUV is about 1/2 of that of caudal fin AUV.

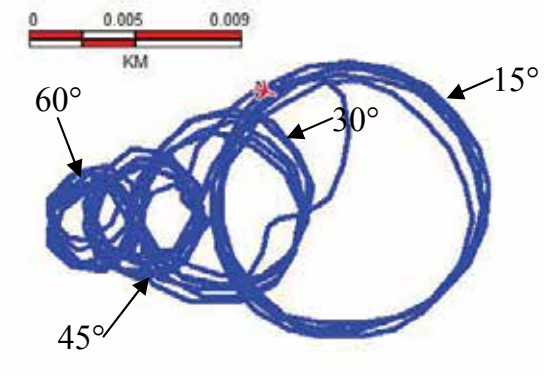


Fig.16 The trajectory of SPC-III AUV performing different caudal fin deflection at about 1.1m/sec.

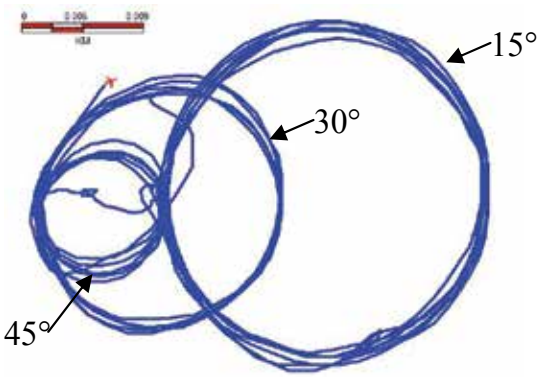


Fig. 17. the trajectory of the Comparison AUV performing different rudder deflection at about 1.1m/sec.

2.4 Probe experiment on blue-green algae

Probe experiment on blue-green algae can be regarded as a commission to inspect propulsion and maneuverability performance of SPC-III. Located in the area of Changjiang Delta Region, Taihu Lake is the major water source of Wuxi. In the summer of 2007, there was a mass breakout of blue-green algae in Taihu Lake, which became the prime environmental issue harassing the local residents and government. In November 2007, carrying Water Quality Multiprobes (HACH D5X), SPC-III successfully performed a probe cruising of about 49km in the water of Taihu Lake and brought back concentration distributing data of blue-green algae. Some of the probe results are shown in Table 2. Areas under heavy pollution are indicated in red in Figure 9.

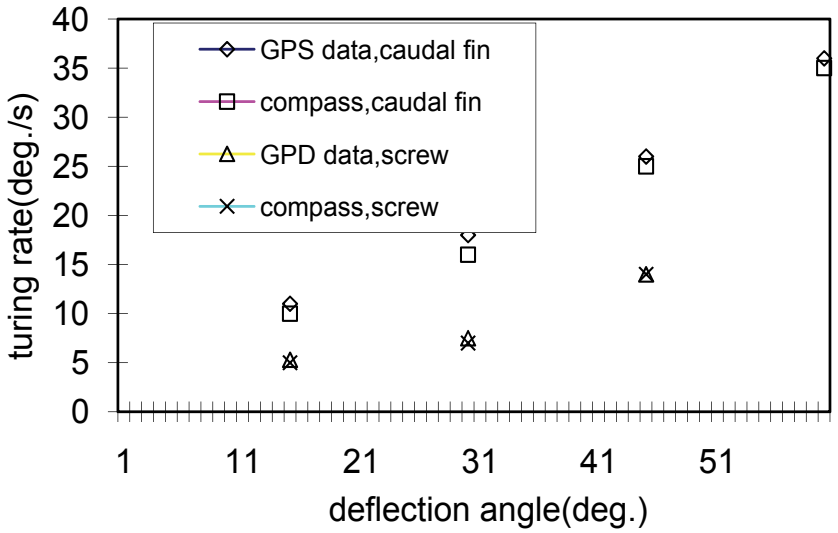


Fig. 18. Yaw rate of SPC-III and the Comparison AUV, at about 1.1m/sec.

Fig. 19. Cruising trajectory of SPC-III at water quality probe on Taihu Lake shown in blue and areas under heavy pollution indicated in red

Data on water quality of Taihu Lake (November,2007)	
average PH value	8.52
maximum PH value	9.51
concentration of blue-green algae (center of the lake)	3823cell/ml.
average pollution concentration (part of lake shore)	288112cell/ml.
Maximum concentration obtained	868120cell/ml.

Table 2. SPC-III brought back data carrying HACH D5X in the water of Taihu Lake

Fig. 20. working environment of SPC-III on Taihu Lake

As a portable UUV, the convenience of SPC-III was proven in the experiment on Taihu Lake. It can be plunged or fished easily by two persons manually without the usage of special ships and devices. Branches and aquatic grass near the bank are often great disaster to small propellers; yet caudal fin thruster which depends on oscillating propulsion can safely pass such area. Thus SPC-III can cruise in water area which is close to the bank and full of aquatic plants. Since blue-green algae are active in these areas, maneuverability advantage of SPC-III is very remarkable. Furthermore, nets or navigation mark often appear on the set navigation route, which requires human intervention to change the course of the vehicle. Nevertheless, relying on greater turning rate, SPC-III can take action when it is very close to the obstacles and does not need early warning. As for the obtruding aquatic bushes it met when cruising in the area a few meters from the bank, SPC-III can steer clear of them with a very small turning radius by slowing down its speed. This is very difficult for a AUV with only one propeller.

Having its batteries charged only one time, SPC-III completed its 49km-mission for 3 days continuously. No default was observed on caudal fin thruster. The reliability of this kind of propeller was preliminarily confirmed.

2.5 Discussions

Compared with high speed dolphin and tuna, the current biorobotics unmanned undersea vehicle still has a long way to go. Yet compared with conventional single-screw propeller AUV, SPC-III has made great progress. With small displacement tonnage, it realizes one-component vector converter and increases low speed maneuverability of AUV remarkably. In addition, the power of caudal fin thruster is also satisfactory. It can be said that using actuating motors to drive two joints caudal fin thruster is a feasible option with the current engineering technology. Of course, there also exist some congenital deficiencies. For example, actuating motors works in oscillation condition and its peak power is 40% higher than that at even pace at similar power output, therefore actuating motor and amplifier possess higher power redundancy. This means power density of the propeller is also reduced. This is the exact reason why Vehicle velocity of SPC-III is hard to increase. Working in oscillation condition also prevents the actuating motors and reducer from work continuously at optimum efficiency points. It is foreseeable that both electro-mechanical conversion efficiency and transmission efficiency of caudal fin thruster are lower than screw

propeller which is in a uniform rotation. In respect of noise, since reducer is adopted, there are no strong points in terms of radiated noise. Yet flapping frequency of caudal fin is far lower than working frequency of the propeller at the same vehicle speed, which means hydrodynamics noise may be low [11]. Future work can be carried out to obtain experiment data on noise through comparison experiment.

2.6 Conclusion

This paper presents an alternative design scheme of two joints caudal fin thruster for portable AUV with single-screw propeller. Using this kind of caudal fin thruster, Biorobotic autonomous undersea vehicle SPC-III has a displacement tonnage of 47kg and a length of 1.75m. The caudal fin thruster only accounts 7% of its displacement tonnage. Comparison experiment on self-propelling has been carried out on the sea. Within the speed of 2~2.7 joints, power consumption of caudal fin thruster and screw propeller is nearly the same. Maximum speed is 1.36m/s and Maximum turning rate is 36°/s. Minimum turning diameter is 2BL, while Minimum turning diameter of the compared propeller AUV is 5BL. Theoretically, Equipped with inside 2352Wh power units, endurance can reach 20 hours at two knots.

3. Reference

YinSheng Zhang, "Underwater Archaeology and Its Exploration Technology", Southeast Culture, no. 4, pp. 29-33, 1996.in chinese.

XiSheng Feng, "From Remotely Operated Vehicles to Autonomous Undersea Vehicles", Engineering Science, vol. 2, no. 12, pp. 29-33, Dec 2000. .in chinese.

JunFeng Huang, et al, "Remote Operated Vehicle(ROV) Dynamic Positioning Based on USBL(Ultra Short Base Line)", Control Engineering of China, vol. 9, no. 6, pp. 75-78, Nov 2002 .in chinese.

"Fish-like swimming, http://www.draper.com/tuna_web/vcuuc.html.

F. E. Fish and J. J. Rohr, "Review of dolphin hydrodynamics and swimming performance," United State Navy Technical Report 1801, Aug.1999.

T. G. Lang, T. Y. Wu, C. J. Brokaw, and C. Brennen, "Speed, power, and drag measurements of dolphins and porpoises." Swimming and Flying in Nature, pp. 553–571, Eds. Plenum Press, New York, NY, 1975.

Oscillating foils of high propulsive efficiency. J. M. Anderson, K. Streitlien et al. [J] Fluid Mech., 1998, 360: 41-72.

Drag Reduction in Fish-like Locomotion. D.S.Barrett, M.S.Triantafyllou, et al. [J] Fluids Mechanics. 1999,392:183-212.

M.S.Triantafyllou, G.S. Triantafyllou, D.K.P. Yue. Hydrodynamics of Fishlike Swimming [J]. Annu. Rev. Fluid Mech. 2000, 32: 33-53

Cheng JY,Zhuang LX,Tong BG.Analysis of swimming three-dimensional waing plate.J Fluid Mech,1991,232:341~355

P. R. Bandyopadhyay, "Trends in biorobotic autonomous undersea vehicles," IEEE J. Oceanic Eng., vol. 30, no. 1, pp. 109–139, Jan. 2005.

J. M. Anderson and N. K. Chhabra, Maneuvering and stability performance of a robotic tuna, Integ. Comp. Biol., vol. 42, 118–126, 2002.

J.M.Anderson and P.A. Kerrebrock. The Vorticity Control Unmanned Undersea Vehicle(VCUUV)-An autonomous vehicle employing fish swimming propulsion and maneuvering [C]. Proc.10th Int. Symp. Unmanned Untethered Submersible Technology. NH, sept, 1997: 189-195

M. Nakashima, K. Tokuo, K. Aminaga, K. Ono. Experimental Study of a Self-Propelled Two-joint Dolphin Robot. Proceedings of the Ninth International Offsore and Polar Engineering Conference. 1999:419-424

M. Nakashima and K. Ono, "Development of a two-joint dolphin robot," in Neurotechnology for Biomimetic Robots, J. Ayers, J. L. Davis, and A.Rudolph, Eds. Cambridge, MA: MIT Press, 2002.

M. Nakashima, Y. Takahashi, T. Tsubaki, and K. Ono, "Threedimensional maneuverability of a dolphin robot (roll control and loop-theloop motion), Proc. of the 2nd International Symposium on Aqua Bio-Mechanisms, 2003, CD-ROM: S.6–10.

Fletcher, B. UUV master plan: a vision for navy UUV development , OCEANS 2000 MTS, 2000:65-71

Liang Jianhong,Wang Tianmiao,Zou Dan,Wang Song,Wang Ye, Trial Voyage of "SPC-II" Fish Robot, transaction of Beihang University,2005,31(7):709-713.

Tianmiao Wang, Jianhong Liang. Stabilization Based Design and Experimental Research of a Fish Robot. the proceeding of IEEE IROS2005,2005: 954 -959

JianHong Liang, TianMiao Wang, Song Wang, Dan Zou , Jian Sun, Experiment of Robofish Aided Underwater Archaeology, the proceeding of IEEE ROBIO2005,2005.

http://www.ifly-uav.com/viewintranews.asp?id=6&menu=news

JianHong Liang, 2006,Propulsive Mechanism of Bionic Undersea Vehicle, Ph.D. Diss., BEIHANG University, Beijing

Liangmei Ying, Jianliang Zhu ,Screw design and implementary on Comparison UUV, Report of CSSRC ,2006

11

Computer Vision Applications in the Navigation of Unmanned Underwater Vehicles

Jonathan Horgan and Daniel Toal
University of Limerick
Ireland

1. Introduction

The inquisitive nature of humans has lead to the comprehensive exploration and mapping of land masses on planet earth, subsequently scientists are now turning to the oceans to discover new possibilities for telecommunications, biological & geological resources and energy sources. Underwater vehicles play an important role in this exploration as the deep ocean is a harsh and unforgiving environment for human discovery. Unmanned underwater vehicles (UUV) are utilised for many different scientific, military and commercial applications such as high resolution seabed surveying (Yoerger *et al.* 2000), mine countermeasures (Freitag *et al.* 2005), inspection and repair of underwater man-made structures (Kondo & Ura 2004) and wreck discovery and localisation (Eustice *et al.* 2005).

Accurate vehicle position knowledge is vital for all underwater missions for correct registration between sensor and navigation data and also for control and final recovery of the vehicle. The characteristics of the underwater environment pose a plethora of difficult challenges for vehicle navigation and these obstacles differ greatly from the issues encountered in land, air and space based navigation (Whitcomb 2000). The rapid attenuation of acoustic and electromagnetic radiation in water restricts the range of acoustic and optical sensors and also limits communication bandwidth. As a consequence of this severe absorption acoustic and optical sensors require submersion near to the survey mission site to gather accurate high resolution data sets. The limitation on communication bandwidth means that vehicle autonomy can only be achieved when the large majority of computation is performed onboard. Whereas land based vehicles can rely on Global Positioning System (GPS) for accurate 3D position updates, the underwater equivalent acoustic transponder network is limited by range, accuracy, the associated cost and deployment & calibration time.

Another challenge that is faced with underwater navigation is the intrinsic ambient pressure. While terrain based vehicle developers have to consider the relatively simplistic and well understood nature of atmospheric pressure in sensor and actuator design, underwater pressure, increasing at a rate of approximately 1 atmosphere (14.7 psi) every 10 meters of depth, can greatly influence and restrict sensor and actuator design. Other issues such as the inherent presence of waves and underwater currents can make the task of accurately describing vehicle motion more difficult and, as a result, affect the accuracy of vehicle navigation.

Many of these problems cannot be overcome directly so the underwater community relies on improving the navigation sensors and the techniques in which the sensor data is interpreted. The development of more advanced navigation sensors is motivated by the need to expand the capabilities and applicability of underwater vehicles and to increase the accuracy, quantity and cost effectiveness of oceanographic data collection. Sensor selection can depend on many factors including resolution, update rate, cost, calibration time, depth rating, range, power requirements and mission objectives. In general the accuracy of a particular sensor is directly proportional to its expense. This has lead to increased research efforts to develop more precise lower cost sensors and improve data interpretation by implementing more intelligent computation techniques such as multi sensor data fusion (MSDF). Many commercially available underwater positioning sensors exist but unfortunately no one sensor yet provides the perfect solution to all underwater navigation needs so, in general, combinations of sensors are employed. The current state of the art navigation systems are based on the use of velocity measurements from a Doppler velocity log (DVL) sensor conveniently fused with accurate velocity/angular rate and position/attitude measurements derived by integration and double integration respectively of linear acceleration and angular rates from an inertial measurement unit (IMU) (Kinsey *et al.* 2006). To bound the inherent integration drift in the system position fixes from an acoustic transponder network such as Long Baseline (LBL), Ultra Short Baseline (USBL) or GPS Intelligent Buoys (GIB) are commonly used. However, this option raises the mission cost as transponders require deployment prior to the mission or a mother ship is necessary. This solution also limits the area in which the vehicle can accurately navigate to within the bounds of the transponder network (acoustic tether).

Over recent years, computer vision has been the subject of increased interest as a result of improving hardware processing capabilities and the need for more flexible, lightweight and accurate sensor solutions (Horgan & Toal 2006). Many researchers have explored the possibility of using computer vision as a primary source for UUV navigation. Techniques for implementing computer vision in order to track cables on the seabed for inspection and maintenance purposes have been researched (Balasuriya & Ura 2002; Ortiz *et al.* 2002). Station keeping, the process of maintaining a vehicle's pose, is another application that has taken advantage of vision system's inherent accuracy and high update rates (Negahdaripour *et al.* 1999; van der Zwaan *et al.* 2002). Motion estimation from vision is of particular interest for the development of intervention class vehicle navigation (Caccia 2006). Wreckage visualization and biological and geological surveying are examples of applications that use image mosaicking techniques to acquire a human interpretable view of the ocean floor but it has also been proven as an appropriate means for near seabed vehicle navigation (Negahdaripour & Xu 2002; Garcia *et al.* 2006).

This chapter gives and introduction to the field of vision based unmanned underwater vehicle navigation and details the advantages and disadvantages of such systems. A review of recent research efforts in the field of vision based UUV navigation is also presented. This review is discussed under the following headings in relation to recent literature reviewed: image mosaicking, cable tracking, station keeping and positioning & localisation. This chapter also considers the applications of sensor fusion techniques for underwater navigation and these are also considered with reference to recent literature. The author gives an opinion about the future of each application based on the presented review. Finally conclusions of the review are given.

2. Underwater optical imaging

Underwater optical imaging has many interesting and beneficial attributes for underwater vehicle navigation, as well as its ability to open up a wealth of understanding of the underwater world. However, it is not an ideal environment for optical imaging as many of its properties inherently affect the quality of image data. While image quality is a pertinent issue for vision system performance, other difficulties are also encountered such as the lack of distinguishable features found on the seafloor and the need for an artificial light source (Matsumoto & Ito 1995). For most UUV applications (below 10 meters) natural lighting is not sufficient for optical imaging so artificial lighting is essential. Light is absorbed when it propagates through water affecting the range of vision systems (Schechner & Karpel 2004). Many variables can affect the levels of light penetration including the clarity of the water, turbidity, depth (light is increasingly absorbed with increasing depth) and surface conditions (if the sea is choppy, more light will be reflected off the surface and less light transmitted to the underwater scene) (Garrison 2004).

Underwater optical imaging has four main issues associated with it: scattering, attenuation, image distortion and image processing. Scattering is as a result of suspended particles or bubbles in the water deflecting photons from their straight trajectory between the light source and the object to be viewed. There are two different types of scattering; backscatter and forward scatter (see Fig. 1). Backscatter is the reflection of light from the light source back to the lens of the camera. This backscattering can result in bright specs appearing on the images, sometimes referred to as marine snow, while also affecting image contrast and the ability to extract features during image processing. Forward scatter occurs when the light from the light source is deflected from its original path by a small angle. This can result in reduced image contrast and blurring of object edges. The affect of forward scatter also increases with range.

The rapid absorption of light in water imposes great difficulty in underwater imaging. This attenuation necessitates the use of artificial lighting for all but the shallowest of underwater missions (less than 10m, dependent on water clarity). The visible spectrum consists of several colours ranging from the red end of the spectrum (wavelength of <780nm) to the blue (wavelength of >390nm). Water effectively works as a filter of light, being more efficient at filtering the longer wavelength end of the visible spectrum, thus absorbing up to 99% of red light by a depth of approximately 4m in seawater (Garrison 2004). Absorption intensifies with increasing depth until no light remains (see Fig. 1). The effects of absorption discussed apply not only to increasing depth but also to distance.

Due to the extreme pressures associated with deep-sea exploration there is need for high pressure housing around each sensor. In the case of a camera a depth rated lens is also required. Imperfections in the design and production of the lens can lead to non-linear distortion in the images. Moreover, the refraction of light at the water/glass and glass/air interface due to the changes in medium density/refractive-index can result in non-linear image deformation (Garcia 2001). To account for this distortion the intrinsic parameters of the camera must be found through calibration and using radial and tangential models the lens distortion effects can be compensated for. The characteristics of the underwater environment not only create issues for collection of clear and undistorted images but also affect the subsequent image processing. Due to the severe absorption of light and the effects of scattering (marine snow etc.) it is essential to decrease range to the objects being viewed in order to obtain higher resolution clearer images. This has the consequence of limiting the

field of view (FOV) of the camera and thus not allowing for wide area images of the seafloor while also challenging the assumption that changes in floor relief are negligible compared to camera altitude.

The motion of the artificial light source attached to the vehicle leads to non-uniform illumination of the scene thus causing moving shadows which makes image to image correspondence more difficult. The lack of structure and unique features in the subsea environment can also lead to difficulties in image matching. While terrestrial applications can make use of man-made structures, including relatively easily defined points and lines, the subsea environment lacks distinguishable features. This is in part due to the lack of man-made structures but also due to the effects of forward scattering blurring edges and points. An issue that affects all real-time image processing applications is whether the hardware and software employed are capable of handling the large amounts of visual data at high speed. This often requires a trade-off in image processing between the frame rate and the image resolution which can be detrimental to the performance of the application.

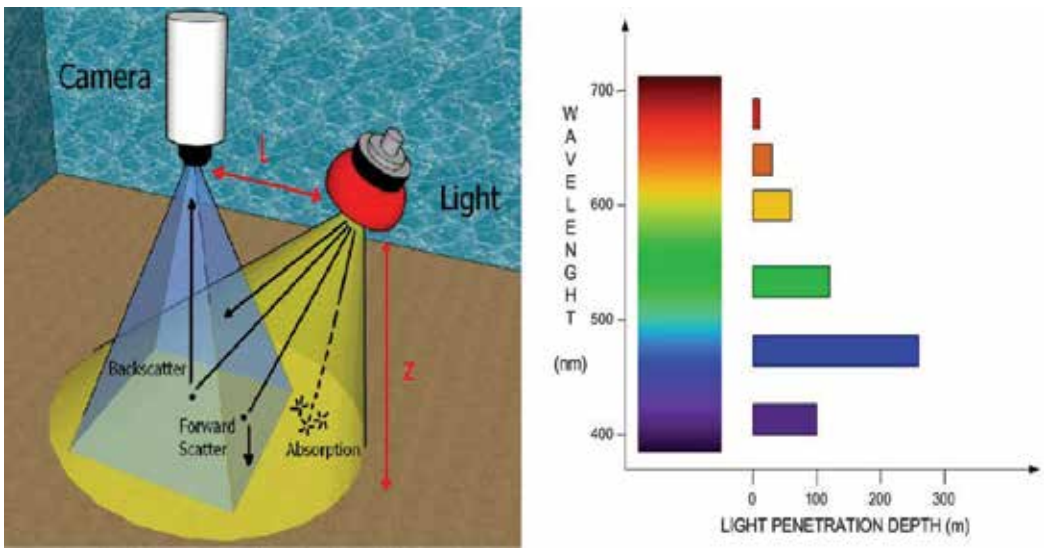


Fig. 1. Scattering and light attenuation (left), colour absorption (right) (Garrison 2004).

3. Vision based navigation

Cameras are found on almost all underwater vehicles to provide feedback to the operator or information for oceanic researchers. Vision based navigation involves the use of one or more video cameras mounted on the vehicle, a video digitizer, a processor and, in general, depending on depth, a light source. By performing image processing on the received frames, the required navigation tasks can be completed or required navigation information can be calculated. The usual setup for the vision system is a single downward facing camera taking images of the sea floor at an altitude of between 1 and 5 meters (see Fig. 2). The use of optical systems, like all navigation sensors, has both advantages and disadvantages. If the challenges of underwater optical imaging, described in section 2, can be successfully addressed some of the potential advantages of vision based underwater navigation include:

- Underwater vehicles are commonly fitted with vision sensors for biological, geological and archaeological survey needs. As such, they have become standard equipment onboard submersibles. As a readily available sensor, vision can be incorporated into a navigation framework to provide alternative vehicle motion estimates when working near the seafloor in relatively clear water.
- The visual data received from optical systems can be easily interpreted by humans and thus provides an effective man-machine interface. Further processing of the visual data can be processed to perform vehicle navigation.
- Optical imaging systems are relatively inexpensive sensors and only require the camera itself, an image digitizer, a host computer and a light source dependent on conditions. The depth rating, low light sensitivity, resolution and whether the camera is zoom or non-zoom, colour or monochrome can all affect pricing.
- Cameras are relatively light weight with small form factors and low power consumption. These can be important issues for deployment on autonomous craft. Unfortunately most missions require artificial lighting which adds significantly to both the weight and power demands of the system.
- Optical imaging has a very high update rate or frame rate and thus allows for high update rate navigation data. The image digitizing hardware and the computation cost of the image processing algorithms are the constraints of the system rather than the optical imager itself.
- Optical imaging systems provide high resolution data with measurement accuracies in the order of millimetres when working near the seafloor.
- Imaging systems can provide 3D position (stereovision) and orientation information, in a fixed world coordinate frame, without requiring the deployment of artificial landmarks or transponders.
- Optical systems have been proven to be capable of providing underwater vehicle navigation without the aid of other sensors.
- Optical imaging systems are very diverse and can be implemented to perform many navigation and positioning applications including: cable tracking, mosaicking, station keeping and motion estimation.

For the purposes of the review the general setup and assumptions about the state of the vehicle and the environment conditions are described. These assumptions are adhered to by all literature and algorithms described unless specifically stated otherwise.

- The underwater vehicle carries a single down-looking calibrated camera to perform seabed imaging.
- The underwater vehicle and thus the camera is piloted at an altitude above the seafloor which allows the acquisition of satisfactory seafloor imagery. This altitude can be affected by external conditions affecting the maximum imaging range.
- The imaged underwater terrain is planar. In most underwater environments this is not the case but the affects of this assumption are reduced using robust statistics for more accurate vehicle motion recovery. This assumption can also be relaxed due the fact that the differences in depth within the imaged seabed are negligible with respect to the average distance from the camera to the seabed.
- The turbidity of the water allows for sufficient visibility for reasonable optical imaging of the working area.
- The light present in the scene is sufficient to allow the camera to obtain satisfactory seafloor imagery.

- An instrumented platform which allows for comparison of results or measurement data fusion is employed.
- Known reference frames between the vehicle and the camera and the vehicle and any other sensors utilized in the navigation technique.

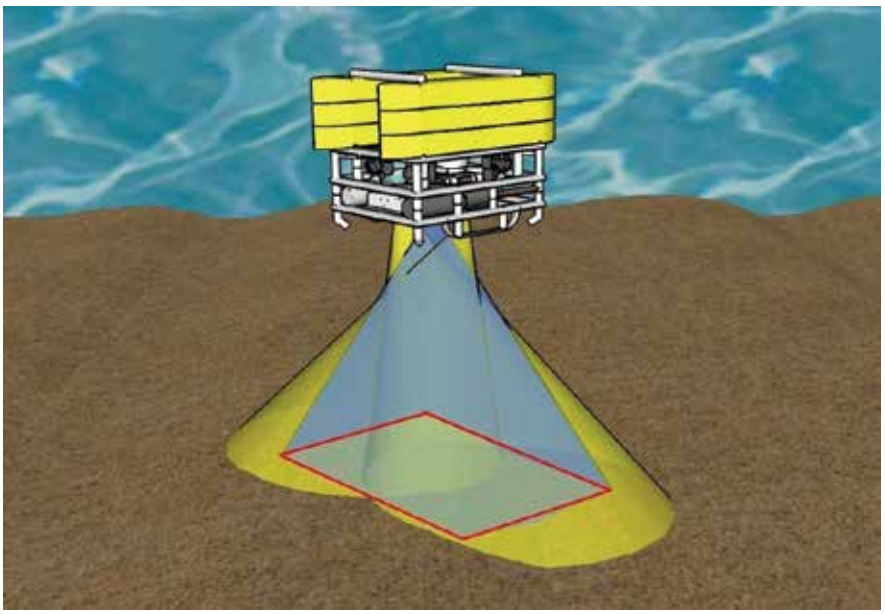


Fig. 2. Camera and lights setup (red box illustrates image frame)

4. Cable tracking

The necessity for frequent underwater cable/pipe inspection is becoming more apparent with increased construction of subsea piping networks for the oil and gas industry and heavy international telecommunication traffic. Current methods for the surveillance, inspection and repair of undersea cables/pipes utilize remotely operated vehicles (ROV) controlled from the surface. This work can prove to be very tedious and time consuming while also being prone to human error due to loss of concentration and fatigue. Cables can be covered along sections of their length thus making it difficult to recover cable trajectory after losing track. A reliable image processing based cable tracking system would prove much less expensive and less prone to error than current solutions, as the need for constant operator supervision is removed. The development of the vision based cable tracking system for use on autonomous vehicles would also be beneficial because of the reduced cost as a mother ship is no longer necessary and such systems are beginning to appear in commercial use (Hydro-International 2008). Vision systems also possess advantages over magnetometer and sonar based solutions for cable tracking (Ito *et al.* 1994). Vision systems prove less expensive, have the ability to identify faults and require a smaller less powerful vehicle for operation (Ortiz *et al.* 2002).

An early attempt at a cable tracking system using machine vision was developed by Matsumoto and Ito (Matsumoto & Ito 1995). The method, like most underwater cable tracking techniques, takes advantage of the lack of straight line edges found in the

underwater environment. An edge image of the sea floor is acquired using a Laplacian of Gaussian filter. The Hough transform is then applied to the edge pixel image in order to find the most likely pipe edge candidates. A method of candidate evaluation is implemented by examining the length and width of each edge pixel line candidate. The direction of the cable in the present image and the previous image are used to predict the angle of the Hough transform to be applied to the subsequent image to reduce computation time. This cable following algorithm also attempts to address the problems of sediment covered pipes and non-uniform illumination. While achieving reasonable results in a controlled environment, factors such as spurious edge detection from other pipes or elements, abrupt pipe direction changes and a search algorithm (when cable is undetected) have not been accounted for and result in reduced performance.

Balisuriya *et al.* developed on previous work (Balasuriya *et al.* 1997) by adding an *a priori* map of the cable location to his technique (Balasuriya & Ura 2002). The main features of the method are the ability to follow the cable when it is not visible to the vision system and selection of the correct cable in the image (in the case of multiple cable presence). These objectives are addressed by assuming that an *a priori* map of the cable is available. The *a priori* map serves three purposes; to predict the region of interest (ROI), to avoid misinterpretations with other cables in the image and to be used as a navigation map in the case where the cable disappears from view. A similar method to Matsumoto and Ito is implemented to locate the cable in the image by utilizing the Hough transform. The technique described fuses inputs other than optical information to track the cable and has attempted to overcome the issues of tracking a cable when it becomes partially or fully obscured to the vision system (due to sediment or algae coverage). It also addresses the difficulty associated with correct cable selection. The method demonstrates that the extra information, in the form of a map, fused with optical sensing can greatly improve performance. Unfortunately, having an *a priori* map of the cable location is not always a realistic assumption especially in the case of older installations.

Ortiz *et al.* developed a method for real-time cable tracking using only visual information that again takes advantage of the cables shape to locate strong alignment features along its side (Ortiz *et al.* 2002). After the initial image segmentation step the contour pixels are examined to locate pixel alignments that display strong pipe characteristics (long pixel alignments, parallel alignments and alignments in a y direction on the image). Once the cable has been located in the image a Kalman filter is implemented to reduce the ROI for the subsequent image to reduce computation time. When anomalies occur in the prediction phase actions are taken in order to correct the algorithm; either the frame causing the anomaly is discarded or, if a number of consecutive frames are incorrect, the Kalman filter is reset. This method achieved a 90 percent success rate for trials at 25 frames/sec performed on old cable installations. The technique dealt reasonably well with partially covered cables however, a minimal presence of the cable is required in the image at all times. No backup system in the scenario where the cable becomes undetectable by the system is described. The performance of the method discussed by Ortiz *et al.* (Ortiz *et al.* 2002) was later improved upon while also reducing the complexity of the system (Antich & Ortiz 2005). This new technique also includes a first approximation to the vehicle control architecture for locating and tracking the cable autonomously using the vision system and a method is proposed for unsupervised tuning of the control system. Both the control system and the tuning strategy were validated using 3D object-oriented simulator implemented in C++ using the OpenGL

graphics library. Only simulation results have been published to date but results for the implemented control architecture are promising.
Recently Wirth *et al.* developed a method for cable tracking by implementing a particle filter in an attempt to predict the location of the cable when it is partially obscured and thus the number of extracted image features is reduced (Wirth *et al.* 2008). A motion model is calculated to describe the cable parameters' changes over time using previously captured cable inspection footage. An observation model is also described to detect cable edges in the image. These models are then combined in a particle filter which sequentially estimates the likelihood of the cable position in subsequent frames. Experimental results concluded that the system was capable of working online in real time and showed good performance even in situations where the cable was scarcely visible. A method for dealing with multiple cable presence has yet to be developed for the system.
Different methods for cable tracking systems exist each with their own advantages and disadvantages. The work reviewed uses similar techniques for cable detection (looking for straight line edges) but differ in their approaches to cable direction prediction to save on computational expense and improve detection robustness. The need for a robust system for tracking a cable that is partially obscured for a short segment remains a priority. Sensor fusion has been proved to be a good approach to robust cable following when the cable is in view. The future will focus on refining tracking methods and working towards the development of vision systems for inspection, fault identification and localisation with the hope of fully automating the process of cable tracking and inspection and reducing human input. There remains a lot of room for improvement in these systems but despite this there is a surprising lack of publications in the field over recent years.

5. Station keeping

The ability for submersible vehicles to accurately maintain position and orientation is a necessity. The process of maintaining a vehicle's predefined pose in the presence of disturbances (undersea currents and reaction forces from manipulators attached to vehicle) is known as station keeping. Station keeping can be used for many different underwater applications such as repair of underwater structures and near seabed data collection. Station keeping using a vision system has the advantage of being able to use natural rather than manmade beacons for motion detection while inherently having a high resolution and update rate. The camera is setup in a similar fashion to that of image mosaicking and the methods for motion estimation overlap greatly between the two applications (see Fig. 2). The general method for visual station keeping is to maintain a reference image acquired from the station and compare live incoming frames with this image to estimate and correct for vehicle drift.
Stanford/MBARI researchers proposed a method of measuring vehicle drift using a texture tracking strategy (Marks *et al.* 1994). The method of motion estimation is the same process as described in video mosaicking (Marks, *et al.*, 1995). Firstly the spatial intensity gradient of the images is filtered to highlight zero crossings using a Laplacian of Gaussian filter. The incoming images are then correlated with the reference image in order to measure movement of features. Filtering in this case is an attempt to highlight image textures and reduce the effect of noise and non-uniform illumination. Tests were performed in a test tank while the vehicle was on the surface but no external measurements were taken in order to thoroughly evaluate the performance of the system. The result consisted of the plots of

commanded control effort to counteract the disturbances in order to hold station. Such a method depends on having a highly textured image in order to find regions of correlation. Correlation-based methods' inability to deal with changes in the image due to rotations will inhibit accurate motion estimation.

Negahdaripour *et al.* proposed a method of station keeping by directly measuring motion from spatio-temporal image gradient information (optical flow) (Negahdaripour *et al.* 1998; Negahdaripour *et al.* 1999). This method allows for the estimation of 3D motion directly using the spatio–temporal derivatives of incremental images captured over a short period of time (Negahdaripour & Horn 1987). A generalized dynamic image motion model was later developed (Negahdaripour 1998) to account for variations in the scene radiance due to lighting and medium conditions underwater. This is of particular importance when using flow–based methods in underwater imagery due to the artificial light source motion. A technique for calculating both instantaneous velocity and absolute position is implemented to increase the limit of inter-frame motion. The position calculated by integrating the velocity over time is used for course correction before the absolute position is used for finer adjustment. This method is susceptible to sporadic miscalculations in velocity, which, accumulated over time, can result in inaccurate position estimations.

Cufi *et al.* (Cufi *et al.* 2002) make use of a technique previously developed for a mosaicking application (Garcia *et al.* 2001b). The acquired images are convolved with high pass filters in both the x and y direction in order to find small windows with the highest spatial gradient (interest points). These windows are then compared to the reference image using two methods. Firstly a correlation based strategy is used to find candidate matches for each interest point. Then a texture characterisation method is performed on each point to select the best correspondence using different configurations of the energy filters (Garcia *et al.* 2001a). As stated above the correlation method is incapable of dealing with large rotations in images due to yaw motion of the vehicle. This problem is overcome in this case by simultaneously creating an image mosaic. The mosaic creation method is based on previous work completed by Garcia *et al.* and is discussed further in section 6 (Garcia *et al.* 2001b). The implementation of the image mosaic also allows for greater inter-frame motion. No overlap between image iterations is needed as the mosaic can be referenced for motion estimation. This method improves on previous correlation based approaches but could again suffer from a lack of distinct textures in the subsea environment while the execution of the mosaicking system may be too computationally expensive to be performed in a real-time on board computer.

Other methods implement a combination of methods to achieve station keeping. Van Der Zwaan *et al.* use a technique of integrating both optic flow information with template matching in order to estimate motion (van der Zwaan *et al.* 2002). The station keeping system tracks an automatically selected naturally textured landmark in the image plane whose temporal deformations are then used to recover image motion. A prediction of the location of the landmark is made by utilizing optical flow information. This estimate is then refined by matching the image with the selected reference frame. This system performed in real-time and showed robust results even in the case of limited image textures however, experiments were performed on poor resolution images thus decreasing accuracy and improving algorithm speed.

Station keeping, much like mosaicking, has many methods for tracking motion from vision: correlation based, feature based, optical flow based etc and selection of the most appropriate

method is by no means a trivial task. Many factors have to be considered to obtain accurate results with the final goal of creating an autonomous real-time station keeping system. The methods discussed are hard to compare due to differing test setups and vehicle dynamics, however, none of the methods mentioned appears fully capable of overcoming the difficulties of station keeping faced in underwater environments, at least in a real-time on board system in an unstructured environment. While improved hardware will allow for the analysis of higher resolution images and thus superior accuracy, there still remains room for algorithm advances and sensor fusion research in order to reproduce the results gained in controlled pool trials and simulations in actual real ocean environments.

6. Mosaicking

Light attenuation and backscatter inhibit the ability of a vision system to capture large area images of the sea floor. Image mosaicking is an attempt to overcome this limitation using a process of aligning short range images of the seabed to create one large composite map. Image mosaicking can be used as an aid to other applications such as navigation, wreckage visualisation, station keeping and also to promote a better understanding of the sea floor in areas such as biology and geology. Mosaicking involves the accurate estimation of vehicle motion in order to accurately position each frame in the composite image (mosaic). The general setup of the vision system remains the same for almost all mosaicking implementations. A single CCD camera is used to acquire images at a right angle to the seabed at an altitude ranging from 1-10 meters depending on water turbidity (see Fig. 2).
One of the very earliest attempts at fusing underwater images to make a larger composite seafloor picture was published by Haywood (Haywood 1986). The simple method described did not take advantage of any image processing techniques but instead used the known vehicle offsets to merge the images in post processing. This method led to aesthetically poor results and gaps in the mosaic. Early attempts at automated image mosaicking were developed by Marks *et al.* who proposed a method of measuring offsets and connecting the images using correlation to create an accurate real-time mosaicking system (Marks *et al.* 1995). This method uses the incoming images to decide the position offset, rather than another type of sensor (acoustic), so it guarantees no gaps are encountered in the mosaic. Much like Marks *et al.* method for station keeping, discussed in the previous section, a stored image is correlated with live incoming images to derive the offset in pixels (Marks *et al.* 1994). The images are filtered using a Laplacian of Gaussian filter in order to highlight zero crossings and pronounce the image textures. The filtering reduces the image noise and also the effect of non-uniform illumination from artificial sources. The mosaic is created by repeatedly storing images and determining by the offset calculated where to place the image in the scene. The images are stored at intervals determined by predefined positional offsets in the x and y planes. Each time an image is stored, the system waits until the x and y value change limit has been reached and the process repeats itself. The system produced was capable of creating single column mosaics in real time using special purpose hardware. This correlation based method relies on well contrasted images in order to locate regions of correlation; a lack of texture will inhibit the system from correctly positioning images in the mosaic. A simple motion model is assumed as correlations inability to deal with rotations, scale changes and undersea currents (seen from results) may hinder its ability to create multiple column mosaics. This method was later extended by Fleisher *et al.* in order to reduce the effect of error growth due to image misalignments, in a similar fashion to current

Simultaneous Localisation and Mapping (SLAM) algorithms (Fleischer 2000). This involved the detection of vehicle trajectory crossover paths in order to register the current images with the stored frames to constrain the navigation error in real time. The use of either an augmented state Kalman filter or a least-squares batch formulation for image realignment estimation was proposed. The same image registration method is implemented thus the system continues to use a simplistic 2D translation image registration model.

Garcia *et al.* proposed a method of feature characterisation to improve the correspondences between images in order to create a more accurate mosaic to position an underwater vehicle (Garcia 2001; Garcia *et al.* 2001b). Firstly regions of high spatial gradient are selected from the image using a corner detector. Image matching is accomplished by taking the textural parameters of the areas selected and correlating them with the next image in sequence. A colour camera improves the process as the matching is implemented on the hue and saturation components of the image as well as the intensity of the image. A set of displacement vectors for the candidate features from one image to the next is calculated. A transformation matrix can then be constructed to merge the images in the correct location in the final mosaic. The paper also implements a smoother filter which is an improvement on techniques first proposed by Fleischer *et al*. (Fleischer 2000). An augmented Kalman filter is used as the optimal estimator for image placement and has the advantage over batch methods of being able to handle multiple loops, real time dynamic optimisation and gives knowledge of the image position variance.

Negadaripour *et al.* extend previously discussed work in station keeping (Negahdaripour *et al.* 1999) and early work in image mosaicking (Negahdaripour *et al.* 1998) to create a fully automatic mosaicking system to aid submersible vehicle navigation (Negahdaripour & Xu 2002). As with the previously discussed station keeping methods, spatio-temporal image gradients are used to measure inter-frame vehicle motion directly which is then integrated over time to provide an estimate of vehicle position. Two methods are proposed for reducing the drift inherent in the system. The first method is based around trying to correct for the biases associated with the optical flow image registration to improve the inter-frame motion estimation and thus reduce accumulated system drift. The second addition attempts to bound the drift in the system by correcting errors in position and orientation at each mosaic update. This is performed by comparing the current image to a region extracted from the mosaic according to the current position estimate. The comparison between the expected image and the current image is used to feedback the correct position estimate and update the mosaic; thus constraining the error growth to the mosaic accuracy.

Gracias *et al.* developed another approach to mosaic creation while also implemented it as an aid for navigation (Gracias 2002; Gracias *et al.* 2003). The estimation of motion is performed by selecting point features on the image using a Harris corner detector (Harris & Stephens 1988) and registering these control points on the proceeding images through a correlation based method. A two step variant of the least median of squares algorithm referred to as the *MEDSERE* is used to eliminate outliers. After estimating the inter-frame motion, the parameters are cascaded to form a global registration where all the frames are mapped to a single reference frame. After registration the mosaic is created by joining the images using the global registration transformation matrix. Where images overlap there are multiple contributions to a single point on the output image. A method of taking the median of the contributors is employed, as it is particularly effective in removing transient data, like moving fish or algae, which has been captured on camera. The creation of the mosaic is

performed offline and then used for real time vehicle navigation. This technique has been experimentally tested for relatively small coverage areas and may not extend well to more expansive surveys due to the assumption of an extended planar scene. The method does not account for lens distortion, which can have a significant impact at larger scales (Pizarro & Singh 2003).

Pizarro *et al.* attempts to tackle the issues associated with the creation of large scale underwater image mosaicking using only image information in a global mosaicking framework (Pizarro & Singh 2003). The problem is broken down in three main parts: radial-distortion compensation, topology estimation and global registration. The proposed method uses feature descriptors invariant to changes in image rotation, scaling and affine changes in intensity and is capable of dealing with low overlap imagery. Radial distortion is accounted for by image warping in a pre-processing step prior to mosaicking. The mosaicking system uses all overlap information, including overlap from images that are not consecutive in time, in order to create a more accurate mosaic by partially limiting the effects of drift. The mosaic is rendered by multi-frequency blending to form a more globally consistent mosaic. The paper claims to have created the largest known published automatically generated underwater mosaic.

Gracias and Neighadaripour present two methods of creating mosaics using video sequences captured at different altitudes (Gracias & Negahdaripour 2005). The first method relies on a rendered mosaic of higher altitude images to act as a map to guide the position of the images in the lower altitude mosaic ('image to mosaic'). The second method does not require rendering of the higher altitude mosaic, just the topology to match each particular image of the lower altitude sequence against the higher altitude images ('image to image'). Ground truth points were used to compare the two methods presented. Both methods obtained good results but while the 'image to image' method showed less distortion, it had the disadvantage of higher computational expense. Unfortunately the method requires a small amount of user input to select correspondences and the flat, static and constant lighting of the environment are assumptions of the technique. Time efficiency is another factor to be considered due to the method requiring runs at different altitudes.

It is difficult to compare and evaluate the performance of each of the methods described. Each technique has been tested in scenarios where different assumptions are made regarding the environment, vehicle dynamics and processing power available. Negahdaripour and Firoozfam attempted to compare methods (using a common data set) implemented by different institutions to document the various approaches and performances of different techniques to the marine world (Negahdaripour & Firoozfam 2001). Unfortunately, due to time constraints, only comparative results for feature–based and direct methods are reported. A more comprehensive report would give a better understanding of the strengths and weaknesses of current techniques available. Some recent research efforts in the area have investigated the construction of 3D mosaics, a further step forward in the evolution of mosaicking methods (Nicosevici *et al.* 2005). Video mosaicking remains a very complex and challenging application because of the inherent difficulties faced with accounting for 3D vehicle motion and the difficulty using optics underwater (Singh *et al.* 2004). 3D mosaicking is a glimpse of what the future could possibly hold for this application and what research institutes will be improving upon with advances in processing capability and vision systems.

7. Positioning & localisation

The possibilities of using vision systems for navigation have already been discussed in the case of mosaicking, station keeping and cable tracking. For the purposes of this review vision based navigation will be discussed in relation to mosaic based localisation, Simultaneous Localisation and Mapping (SLAM) and motion estimation.

Image mosaics are a large area composite view of the seafloor. This composite view is effectively a map of the area over which the vehicle has passed during the mission. If the mosaic updates in real-time and thus the most recent visual information is available it allows for comparison between current camera frames and the composite image in order to improve the mosaic but also to localise the vehicle within the composite image. This technique has been used in both station keeping and mosaicking. Cufi *et al*. compare the live image with the most recently updated mosaic to allow for greater inter-frame motion and improve the robustness of the station keeping system (Cufi *et al.* 2002). Gracias *et al.* used a technique in which the mosaic is created offline and then implemented for a subsequent mission as a map of the site to aid vehicle navigation (Gracias 2002; Gracias *et al.* 2003). Negadaripour and Xu take advantage of the mosaic by calculating the inter-frame motion in order to estimate vehicle position and subsequently use the rendered mosaic to improve the placement of image at the mosaic update stage (Negahdaripour & Xu 2002).

Simultaneous Localisation and Mapping (SLAM) also known as concurrent mapping and localisation (CML) is the process in which a vehicle, starting at an unknown location in an unknown environment, incrementally builds a map within the environment while concurrently using the map to update is current position. Following vehicle motion, if at the next iteration of map building the measured distance and direction travelled has a slight inaccuracy than any features being added to the map will contain corresponding errors. If unchecked, these positional errors build cumulatively, grossly distorting the map and therefore the robot's ability to know its precise location. There are various techniques to compensate for this such as recognising features that it has come across previously and re-skewing recent parts of the map to make sure the two instances of that feature become one. The SLAM community has focused on optimal Bayesian filtering and many techniques exist including laser range scanning (Estrada *et al.* 2005) , sonar (Tardos *et al.* 2002) and video (Davison *et al.* 2007) . Almost all the literature is based on terrestrial environments where vehicle dynamics are more limited and manmade structures provide an abundance of robust scene features. Very little literature exists which has tackled the issues of SLAM based navigation in an underwater environment. The strong majority of research that has taken place in the underwater environment has focused on acoustic data (Tena Ruiz *et al.* 2004; Ribas *et al.* 2006). The key to successful visual SLAM for underwater vehicle navigation lies in the selection of robust features on the sea floor to allow for accurate correspondence in the presence of changing view points and non uniform illumination. Another important factor to be considered is the likely sparseness of image points due to the environment and the necessary selection of robust features.

One of the few examples of underwater optical SLAM was developed by Eustice who implemented a vision based SLAM algorithm that performs even in the cases of low overlap imagery (Eustice 2005). Inertial sensors are also taken advantage of in the technique developed to improve the production of detailed seabed image reconstructions. Using an efficient sparse information filter the approach scales well to large-scale mapping in testing where an impressive image mosaic of the RMS Titanic was constructed (Eustice *et al.* 2005).

Williams *et al.* describes a method of underwater SLAM that takes advantage of both sonar and visual information for feature extraction in reef environments (Williams & Mahon 2004). Unfortunately the performance of the system during testing is difficult to evaluate as no ground truth was available for comparison. Saez *et al.* detail a technique for visual SLAM that takes advantage of a trinocular stereo vision (Saez *et al.* 2006). A global rectification strategy is employed to maintain the global consistency of the trajectory and improve accuracy. While experiments showed good results all testing was carried out offline. The algorithm for global rectification becomes increasingly computational complex with time and as a result is unsuitable for large scale environments. Petillot *et al.* presents an approach to perform underwater 3D reconstruction of the seabed aided by SLAM techniques and the use of a stereo camera system (Petillot *et al.* 2008). A Rauch-Tung-Striebel (RTS) smoother is used to improve the trajectory information outputted by the implemented Kalman filter. This paper is unique in the way it uses a combination of SLAM and RTS techniques for the optical 3D reconstruction of the seabed.

The issues associated with metric motion estimation from vision are dealt with more directly by Caccia (Caccia 2003) and later developed into a more complete system with ocean environment experimental results (Caccia 2007). The system is based on an optical feature correlation system to detect motion between consecutive camera frames. This motion is converted into its metric equivalent with the implementation of a laser triangulation scheme to measure the altitude of the vehicle (Caccia 2006). The current system only allows for horizontal linear translation and doesn't account for changes in yaw but promising results were achieved using the Romeo vehicle for a constant heading and altitude in the Ligurian Sea. Cufi also calculates direct metric motion estimation for evaluation of a station keeping algorithm (Cufi *et al.* 2002). This technique uses altitude measurements gained from ultrasonic altimeter to convert offsets from images produced by a calibrated camera into metric displacements.

Machine vision techniques have been proven as a viable localisation and motion sensor in an unstructured land setting; unfortunately it is by no means a trivial task to transfer these techniques to subsea systems. The underwater environment adds the complexity of 3D motion and the inherent difficulties associated with optics underwater. However, recent work in the area of vision based SLAM and motion estimation techniques have proved that imaging systems can be complementary sensor to current sonar and inertial motion estimation solutions with the advantages of having high accuracy and update rate and being especially beneficial in near intervention environments. The SLAM community is focused on improving algorithms to allow for real time mapping of larger environments while improving robustness in the case of sparse features, changing illumination and highly dynamic motion.

8. Navigation using sensor fusion

Sensor fusion, also known as multi-sensor data fusion (MSDF), is the combination of sensory data or data derived from sensory data from different sources in order to achieve better information than would be possible when these sources are used individually. The term better in this case refers to the data and can mean: more accurate, noise tolerant, more complete, sensor failure tolerant or data with reduced uncertainty. There are many different issues that require consideration when performing sensor fusion such as data alignment, data association, fusion, inference and sensor management (Loebis *et al.* 2002). The fusion

process can also be further categorized by the different levels at which it can take place. A commonly used categorisation identifies four fusion levels signal, pixel, feature and symbol and these are discussed in more detail in references (Loebis *et al.* 2002; Luo *et al.* 2002).

All sensors available for underwater vehicle navigation have their own advantages and disadvantages. Sensor fusion techniques allow for the fusion of data from many sources to improve the overall navigation accuracy and reliability while taking advantage of the available sensors complementary attributes. A well established sensor fusion application is between a Doppler velocity log (DVL) and an inertial navigation system (INS) (Kinsey *et al.* 2006). This sensor fusion is used to combat the issue of INS integration drift: small errors in the measurement of acceleration and angular velocity are integrated into progressively larger errors in velocity, which is compounded into still greater errors in position. Inertial measurement units (IMU) typically employ another type of sensor such as DVL measurements and position measurements from GPS or acoustic navigation systems to correct for errors in the IMU state estimate and limit the effect of integration drift. Whitcomb *et al.* reported preliminary results from the first deployment of an early prototype of a combined long base line (LBL) acoustic network positioning and Doppler navigation (Whitcomb *et al.* 1999). This system was later extended upon by Kinsey *et al.* who indentified that solutions and experimental results for underwater vehicle navigation in the x and y horizontal plane were particularly rare in literature (Kinsey & Whitcomb 2004). The system developed, *DVLNAV* supports many of the sensors available on today's UUVs including DVL, LBL, compass, depth sensors, altimeters and GPS. Results demonstrated that the system provides more accurate navigation at a higher precision and update rate than LBL alone while also proving that accurate estimates of sound velocity, heading, and attitude data in computing underwater vehicle position significantly improves the accuracy of Doppler based navigation.

Loebis *et al.* published a review of MSDF and its application to UUV navigation (Loebis *et al.* 2002). It was concluded that accurate navigation cannot be performed by one navigation system alone and the best way to improve is by implementing MSDF between a number of complementary navigation systems. A method of cable tracking that utilizes MSDF between an INS, GPS, and vision based dead reckoning is also proposed but to the authors' knowledge no results for the system have been published to date. Nicosevici *et al.* presented a classification of currently used sensor fusion techniques that focuses on its application to UUV navigation (Nicosevici *et al.* 2004). Many of systems reviewed implement the extended Kalman filter for sensor data fusion. The main conclusions drawn from the literature for sensor fusion implementation is to first be aware of the goal of the sensor fusion (the improvement brought by the system) and second be aware of the constraints imposed by the sensors involved (sensor data model etc.).

While vision based sensor fusion techniques are growing in popularity for terrestrial robot navigation applications (Dong-Xue et al. 2007; Jia *et al.* 2008) very little literature exists for underwater vision based sensor fusion as most of the navigation applications reviewed rely purely on optical information (Eustice 2005). One of the few underwater vision based sensor fusion techniques is proposed by Balasuriya *et al.* to tackle the issues of cable tracking when the cable becomes invisible to the camera for a short period of time and also correct cable selection in the presence of multiple possibilities (Balasuriya & Ura 2001). A combination of image data, a prior map of the cable location and inertial data are fused together in order to implement reliable cable tracking. Testing of the algorithm using the Twin-Burger 2 AUV in

a test tank proved that the sensor fusion greatly improved system performance. Majunder *et al.* describe an algorithm that takes advantage of low level sensor fusion techniques in order to provide a more robust scene description by combining both vision and sonar information (Majumder *et al.* 2001). Huster *et al.* propose a system to improve station keeping by using accelerometer and gyrocompass measurements as well as monocular vision displacements to counteract drift from a fixed location. The use of inertial measurements also reduces the amount of visual information required for extraction from the vision system resulting in a more simple and robust solution (Huster *et al.* 2002).While vision based motion estimation techniques rely on the fusion of altitude measurements from sensors to estimate metric displacement (Cufi *et al.* 2002), Eustice (Eustice 2005) also takes advantage of other sensor information (attitude) in order to overcome many of the challenging issues involved in visual SLAM based navigation in an unstructured environment.
The authors of this chapter, Horgan *et al.* propose a real-time navigation system for a UUV that takes advantage of the complementary performance of a sensor suite including a DVL, a compass, a depth sensor and altimeter sensors with a feature based motion estimator using vision (Horgan *et al.* 2007). The compass and the depth sensors are used to bound the drift of the heading and depth estimations respectively. The altimeter is required in order to translate the feature displacements measured from the images into the metric displacements of the robot. While the robot must rely on DVL navigation above a certain altitude where vision is less effective, DVL measurements can be complemented with higher frequency accurate motion estimates from the vision system when navigating close to the seafloor. When a vehicle comes close to the seabed, DVL can drop out due to minimum blanking range, however at such short ranges vision systems are at their most effective.
From the reviewed papers it is apparent that sensor fusion can greatly improve robot navigation accuracy while also decreasing the need for expensive individual sensors. However, there is a relative lack of publications in the area which can be explained by the fact that sensor fusion can be quite difficult to implement due to sensors having different physical properties, data types, update rates and resolutions. MSDF that takes advantage of visual information is an appealing prospect as it has complementary attributes to many commercially available sonar sensors as vision system's performance improves with decreasing range making it a very good candidate for near intervention underwater missions. Very little research has taken place into fusion between inertial and vision measurements but the author believes that vision is a viable solution for aiding INS in a near seabed environment where acoustic positioning may be prone to inaccuracy e.g. in channels, caves or wrecks.

9. Conclusion

The presented review illustrates the growing popularity of vision based UUV navigation methods. The diversity of optical imaging has been demonstrated with regard to applications of cable tracking and inspection, mosaicking, positioning & localisation and sensor fusion. Vision systems are a very useful sensor for the navigation of underwater vehicles and have been the subject of increased interest over the last decade as a result of improved processing capabilities of hardware and the need for more flexible and accurate sensor solutions. The increase in research efforts into vision systems is due to its inexpensive nature and its common inclusion on underwater vehicles as a payload sensor. Vision also has a number of advantages over other types of sensors for underwater applications.

Cameras are light weight and do not possess a minimum operating range unlike their acoustic counterparts (Nolan 2006). Despite these advantages over other sensors, machine vision underwater poses an amount of difficult challenges to be overcome for it to be successfully incorporated into control (Matsumoto & Ito 1995). Marine snow, low contrast, non-uniform illumination and a lack of distinguishable features on the seabed are just some of the inherent difficulties faced when using optics underwater (Cufi *et al.* 2002).
Kinsey *et al.* indentified that solutions and experimental results for underwater vehicle navigation in the x and y horizontal plane were particularly rare in literature (Kinsey & Whitcomb 2004). This is of particular interest as vision is a very useful sensor for horizontal plane navigation in the correct conditions while near the seafloor. It was later stated by Kinsey *et al.* that there is a distinct need for improved near seabed (near intervention) UUV navigation systems for the exploitation of scientific data and the near seabed operations (Kinsey *et al.* 2006). Current systems while sufficiently precise and fast for dynamic positioning remain unconvincing in near intervention operations. Vision systems have inherently fast frame rates (update rates) depending on the capability of both the hardware and software while also having resolutions of sub centimetre accuracy depending on vehicle altitude and resolution. These attributes as well as the other advantages discussed previously make vision based methods ideal for near intervention class missions.
Sensor fusion techniques will allow for more complementary synthesis of vision algorithms with DVL, INS, compass and altimeters proven to provide a more robust navigation solution. While online processing of optical data has been an issue in the past improving hardware capabilities should now allow for real-time implementation of vision based algorithms with sufficient accuracy and update rates.

10. References

Antich, J. & Ortiz, A. (2005). Development of the Control Architecture of a Vision-Guided Underwater Cable Tracker. *International Journal of Intelligent Systems* 20(5): 477-498.

Balasuriya, A.; Takai, M.; Lam, W. C.; Ura, T. & Kuroda, Y. (1997). Vision based autonomous underwater vehicle navigation: underwater cable tracking. *OCEANS '97. MTS/IEEE.*

Balasuriya, A. & Ura, T. (2001). On-board sensor fusion scheme for autonomous underwater vehicle navigation in submarine cable inspection. *Multisensor Fusion and Integration for Intelligent Systems, 2001. MFI 2001.*

Balasuriya, A. & Ura, T. (2002). Vision-based underwater cable detection and following using AUVs. *Oceans '02 MTS/IEEE.*

Caccia, M. (2003). Vision-based linear motion estimation for unmanned underwater vehicles. *IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA '03.*

Caccia, M. (2006). Laser-Triangulation Optical-Correlation Sensor for ROV Slow Motion Estimation. *IEEE Journal of Oceanic Engineering* 31(3): 711-727.

Caccia, M. (2007). Vision-based ROV horizontal motion control: Near-seafloor experimental results. *Control Engineering Practice* 15(6): 703-714.

Cufi, X.; Garcia, R. & Ridao, P. (2002). An approach to vision-based station keeping for an unmanned underwater vehicle. *IEEE/RSJ International Conference on Intelligent Robots and System, 2002.*

Davison, A. J.; Reid, I. D.; Molton, N. D. & Stasse, O. A. S. O. (2007). MonoSLAM: Real-Time Single Camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(6): 1052-1067.

Dong-Xue, Y.; Xin-Sheng, H. & Hong-Li, T. (2007). INS/VNS Fusion based on unscented particle filter. *International Conference on Wavelet Analysis and Pattern Recognition, 2007. ICWAPR '07.* .

Estrada, C.; Neira, J. & Tardos, J. D. (2005). Hierarchical SLAM: Real-Time Accurate Mapping of Large Environments. *IEEE Transactions on Robotics* 21(4): 588-596.

Eustice, R. M.; Singh, H.; Leonard, J.; Walter, M. & Ballard, R. D. (2005). Visually navigating the RMS Titanic with SLAM information filters. *Proceedings of Robotics Science and Systems*. Cambridge, MA, MIT Press**:** 57-64.

Eustice, R. M. (2005). *Large-Area Visually Augmented Navigation for Autonomous Underwater Vehicles*. Boston, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution. Doctor of Philosophy**:** 187.

Fleischer, S. D. (2000). *Bounded-error vision-based navigation of autonomous underwater vehicles*. Department of Aeronautics and Astronautics, Stanford University. Doctor of Philosophy**:** 209.

Freitag, L.; Grund, M.; Von Alt, C.; Stokey, R. & Austin, T. (2005). A Shallow Water Acoustic Network for Mine Countermeasures Operations with Autonomous Underwater Vehicles. *Underwater Defense Technology (UDT)*.

Garcia, R.; Xevi, C. & Battle, J. (2001a). Detection of matchings in a sequence of underwater images through texture analysis. *Proceedings of the International Conference on Image Processing*.

Garcia, R. (2001). *A Proposal to Estimate the Motion of an Underwater Vehicle through Visual Mosaicking*. Department of Electronics, Informatics and Automatiuon. Girona, University of Girona. Doctor of Philosophy**:** 187.

Garcia, R.; Batlle, J.; Cufi, X. & Amat, J. (2001b). Positioning an underwater vehicle through image mosaicking. *IEEE International Conference on Robotics and Automation*

Garcia, R.; Cufi, X.; Ridao, P. & Carreras, M. (2006). Constructing Photo-mosaics to Assist UUV Navigation and Station-keeping (Chapter 9). *Robotics and Automation in the Maritime Industries***:** 195-234.

Garrison, T. (2004). *Oceanography: An Invitation to Marine Science*, Thomson Brooks/Cole.

Gracias, N. (2002). *Mosaic–based Visual Navigation for Autonomous Underwater Vehicles*. Instituto Superior T'ecnico Lisbon, Universidade Técnica de Lisboa. Doctor of Philosophy**:** 138.

Gracias, N.; van der Zwaan, S.; Bernardino, A. & Santos-Victor, J. (2003). Mosaic-based navigation for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering* 28(4): 609-624.

Gracias, N. & Negahdaripour, S. (2005). Underwater Mosaic Creation using Video sequences from Different Altitudes. *Proceedings of MTS/IEEE OCEANS*

Harris, C. & Stephens, M. (1988). A combined corner and edge detector. *4th Alvey Vision Conference*: 147-151.

Haywood, R. (1986). Acquisition of a Micro Scale Photographic Survey Using an Autonomous Submersible. *OCEANS*, New York, USA.

Horgan, J. & Toal, D. (2006). Vision Systems in the Control of Autonomous Underwater Vehicles. *7th IFAC Conference on Manoeuvering and Control of Marine Craft (MCMC 2006)*

Horgan, J.; Toal, D.; Ridao, P. & Garcia, R. (2007). Real-time vision based AUV navigation system using a complementary sensor suite. *IFAC Conference on Control Applications in Marine Systems (CAMS'07)*.

Huster, A.; Frew, E. W. & Rock, S. M. (2002). Relative position estimation for AUVs by fusing bearing and inertial rate sensor measurements. *Oceans '02 MTS/IEEE*.

Hydro International (2008). World Record for Autonomous Pipeline Tracking. *Hydro International*, Reed Business. 12**:** 9.

Ito, Y.; Kato, N.; Kojima, J.; Takagi, S. A. T. S.; Asakawa, K. A. A. K. & Shirasaki, Y. A. S. Y. (1994). Cable tracking for autonomous underwater vehicle. *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology, AUV '94.*

Jia, Z.; Balasuriya, A. & Challa, S. (2008). Sensor fusion-based visual target tracking for autonomous vehicles with the out-of-sequence measurements solution. *Robotics and Autonomous Systems* 56(2): 157-176.

Kinsey, J. C. & Whitcomb, L. L. (2004). Preliminary field experience with the DVLNAV integrated navigation system for oceanographic submersibles. *Control Engineering Practice* 12(12): 1541-1549.

Kinsey, J. C.; Eustice, R. M. & Whitcomb, L. L. (2006). Survey of underwater vehicle navigation: Recent advances and new challenges. *IFAC Conference of Manoeuvring and Control of Marine Craft (MCMC 06')*.

Kondo, H. & Ura, T. (2004). Navigation of an AUV for investigation of underwater structures. *Control Engineering Practice* 12(12): 1551-1559.

Loebis, D.; Sutton, R. & Chudley, J. (2002). Review of multisensor data fusion techniques and their application to autonomous underwater vehicle navigation. *Journal of Marine Engineering and Technology* 1: 3-14

Luo, R. C.; Chih-Chen, Y. & Kuo Lan, S. (2002). Multisensor fusion and integration: approaches, applications, and future research directions. *IEEE Sensors Journal* 2(2): 107-119.

Majumder, S.; Scheding, S. & Durrant-Whyte, H. F. (2001). Multisensor data fusion for underwater navigation. *Robotics and Autonomous Systems* 35(2): 97-108.

Marks, R. L.; Wang, H. H.; Lee, M. J. & Rock, S. M. (1994). Automatic visual station keeping of an underwater robot. *OCEANS '94. 'Oceans Engineering for Today's Technology and Tomorrow's Preservation.' Proceedings* 2: 137-142.

Marks, R. L.; Rock, S. M. & Lee, M. J. (1995). Real-time video mosaicking of the ocean floor. *IEEE Journal of Oceanic Engineering* 20(3): 229-241.

Matsumoto, S. & Ito, Y. (1995). Real-time vision-based tracking of submarine-cables for AUV/ROV. *OCEANS '95. MTS/IEEE. 'Challenges of Our Changing Global Environment'*.

Negahdaripour, S. & Horn, B. (1987). Direct Passive Navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 9(1).

Negahdaripour, S. (1998). Revised definition of optical flow: integration of radiometric and geometric cues for dynamic scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(9): 961-979.

Negahdaripour, S.; Xu, X.; Khamene, A. & Awan, Z. A. A. Z. (1998). 3-D motion and depth estimation from sea-floor images for mosaic-based station-keeping and navigation of ROVs/AUVs and high-resolution sea-floor mapping. *Proceedings Of The 1998 Workshop on Autonomous Underwater Vehicles, AUV'98.* .

Negahdaripour, S.; Xu, X. & Jin, L. (1999). Direct estimation of motion from sea floor images for automatic station-keeping of submersible platforms. *IEEE Journal of Oceanic Engineering* 24(3): 370-382.

Negahdaripour, S. & Firoozfam, P. (2001). Positioning and photo-mosaicking with long image sequences; comparison of selected methods. *MTS/IEEE OCEANS, 2001.*

Negahdaripour, S. & Xu, X. (2002). Mosaic-based positioning and improved motion-estimation methods for automatic navigation of submersible vehicles. *IEEE Journal of Oceanic Engineering* 27(1): 79-99.

Nicosevici, T.; Garcia, R.; Carreras, M. & Villanueva, M. (2004). A review of sensor fusion techniques for underwater vehicle navigation. *OCEANS '04. MTTS/IEEE TECHNO-OCEAN '04*.

Nicosevici, T.; Negahdaripour, S. & Garcia, R. (2005). Monocular-Based 3-D Seafloor Reconstruction and Ortho-Mosaicing by Piecewise Planar Representation. *MTS/IEEE OCEANS 2005*.

Nolan, S. (2006). *A High Frequency Wide Field of View Ultrasonic Sensor for Short Range Collison Avoidance Applications on Intervention Class Underwater Vehicles*. Electronic and Computer Engineering. Limerick, University of Limerick. Doctor of Philosophy**:** 177.

Ortiz, A.; Simo, M. & Oliver, G. (2002). A vision system for an underwater cable tracker. *International Journal of Machine Vision and Applications* 13(3): 129-140.

Petillot, Y.; Salvi, J. & Batlle, J. (2008). 3D Large-Scale Seabed Reconstruction for UUV Simultaneous Localization and Mapping. *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV 08')*.

Pizarro, O. & Singh, H. (2003). Toward large-area mosaicing for underwater scientific applications. *IEEE Journal of Oceanic Engineering* 28(4): 651-672.

Ribas, D.; Ridao, P.; Neira, J. & Tardos, J. D. (2006). SLAM using an Imaging Sonar for Partially Structured Underwater Environments. *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Saez, J. M.; Hogue, A.; Escolano, F. & Jenkin, M. (2006). Underwater 3D SLAM through entropy minimization. *Proceedings 2006 IEEE International Conference on Robotics and Automation, ICRA 2006*.

Schechner, Y. Y. & Karpel, N. (2004). Clear underwater vision. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004.*

Singh, H.; Howland, J. & Pizarro, O. (2004). Advances in large-area photomosaicking underwater. *IEEE Journal of Oceanic Engineering* 29(3): 872-886.

Tardos, J. D.; Neira, J.; Newman, P. M. & Leonard, J. J. (2002). Robust Mapping and Localization in Indoor Environments using Sonar Data. *The International Journal of Robotics Research* 21(4): 311-330.

Tena Ruiz, I.; de Raucourt, S.; Petillot, Y. & Lane, D. M. (2004). Concurrent mapping and localization using sidescan sonar. *IEEE Journal of Oceanic Engineering* 29(2): 442-456.

van der Zwaan, S.; Bernardino, A. & Santos-Victor, J. (2002). Visual station keeping for floating robots in unstructured environments. *Robotics and Autonomous Systems* 39(3-4): 145-155.

Whitcomb, L.; Yoerger, D. & Singh, H. (1999). Advances in Doppler-based navigation of underwater robotic vehicles. *IEEE International Conference on Robotics and Automation, 1999.*

Whitcomb, L. L. (2000). Underwater robotics: out of the research laboratory and into the field. *IEEE International Conference on Robotics and Automation, ICRA '00.*

Williams, S. & Mahon, I. (2004). Simultaneous localisation and mapping on the Great Barrier Reef. *IEEE International Conference on Robotics and Automation, ICRA '04.*

Wirth, S.; Ortiz, A.; Paulus, D. & Oliver, G. (2008). Using Particle Filters for Autonomous Underwater Cable Tracking. *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV 08')*.

Yoerger, D.; Bradley, A.; Walden, B.; Cormier, M. H. & Ryan, W. (2000). Fine-scale seafloor survey in rugged deep-ocean terrain with an autonomous robot. *IEEE International Conference on Robotics and Automation, ICRA '00.*

12

AUV Application for Inspection of Underwater Communications

Alexander Inzartsev and Alexander Pavin
Institute of Marine Technology Problems (IMTP FEB RAS)
Far Eastern Branch of the Russian Academy of Sciences
Russia

1. Introduction

Regular inspection of underwater communications (pipelines and cables) is actual problem of modern oil and gas industry. Specially equipped vessels, towed underwater devices and remote operated vehicles /ROV/ are applied for these purposes as usually, but quality of acquired data does not always allow revealing emergencies at the proper time. "Spot" inspections by ROVs give difficultly comparable data (Baker, 1991; Murray, 1991). The perspective solution of the problem is autonomous underwater vehicles /AUV/ application as "the intellectual carrier" of research equipment (Evans et al., 2003; Kojima et al., 1997).
According (Ageev, 2005) the main goals of pipeline and cables inspection are:

1. more accurate position determination (searching and tracking);
2. pipe sagging and freespan detection and measurement;
3. terrain survey on each side of communication by means of high frequency side scan sonar /HF SSS/ and detection of extraneous objects;
4. detection of damages;
5. leakage detection of transported substances (for pipelines).

The pipeline and cable inspection by means of AUV includes two stages: preliminary (communication search and detection) and the main (motion along the communication with carrying out of necessary measurements, i.e. tracking). Exact mutual orientation of AUV and inspected object is required in real time during the tracking stage.
To solve inspection tasks AUV should be equipped with reliable detection systems for inspected object recognition. Video, electromagnetic and echo-sounder data can be used for these purposes. Each of these devices demonstrates optimal results for certain classes of objects in appropriate conditions. For example, metal pipelines have the significant sizes and can be detected by all listed above devices. While underwater cables have a small diameter, because of this applicability of acoustic methods is limited (Petillot et al., 2002). Process of communications search and detection is complicated, as a rule, with a poor visibility of the given objects (strewed with a ground, silted or covered by underwater flora and fauna).
Experiments with the use of AUV for inspection of underwater communications have been carried out for a long time. Usually only one instrument, which AUV is equipped with, is used for object detection.

The first experiments with detection and inspection of metal cables were carried out with the use of AUV "Aqua Explorer 2" (Asai at al., 2000). The AUV was equipped with two external magnetometers which allowed to find out metal cables. However the devices allowed to find out only active cables (i.e. cables with electric current). Devices did not find out cables where there was not electric current for any reasons (for example, because of breakage). Moreover AUV had significant transverse dimensions because of magnetometers installation on pylons.
The basic possibility of video camera use for automatic cables detection and tracking was shown in (Matsumoto & Ito, 1995; Ortiz at al., 2000). And the firsts practical results were obtained with the use of semi-AUV TSL (Scherbatyuk at al., 2000) and ROVs Ventana and Tiburon (Kogan at al., 2006).
The experiments of pipelines tracking with the use of multi-beam echo sounder were carried out on the base of AUV AUTOTRACKER (Petillot et al., 2002) and revealed good enough results. However the used facilities allowed to detect the inspected object only at strict mutual orientation of AUV and the pipeline.
Detection reliability of lengthy metallic objects (cables and pipelines) can be considerably increased by means of synchronous data processing of the all recognition devices. In other words, it is necessary to equip AUV with a plenty of detection systems which work on the basis of different physical principles. For the sake of reliability the information from the different sources is combined into "environment model" and processed jointly. "Environment model" is updated on the basis of new data and can be used for subsequent AUV motion planning. The questions of object recognition, "environment model" structure and AUV behavior during inspection are considered in this paper.

2. Pipeline & cable recognition systems

Further we assume that AUV is equipped with several diverse recognition systems, capable to work simultaneously. For inspected object position estimation all data from recognition subsystems should be in uniform format. It is supposed, that each object identification subsystem n (n=1…N, N – number of subsystems) periodically gives out the following data:

- vector of position and direction of the inspected object (in absolute coordinate system) $\boldsymbol{r}_t^{<n>}=[X_t^{<n>},Y_t^{<n>},\Phi_t^{<n>}]^{\mathrm{T}}$.
- estimation of object existence probability $p_t^{<n>}$.

It is also supposed that AUV is equipped with onboard navigation system which is necessary for recognition subsystems data integration. It gives current AUV coordinates in the next format: $\boldsymbol{r}_t^{\mathrm{AUV}}=[X_t^{\mathrm{AUV}},Y_t^{\mathrm{AUV}},\Phi_t^{\mathrm{AUV}}]^{\mathrm{T}}$. During inspection vehicle moves on a relatively small distance over the sea bottom, therefore a Doppler velocity log /DVL/ can be used for AUV position determination. The best models of modern DVL provide measurement accuracy of speed at a level 1-2 mm/s. The accumulating position error in this case does not exceed 10 meters per hour that is quite allowable for maintenance of a positive control for cable tracking.
For the sake of definiteness we shall assume, that AUV is equipped with the following systems:

- video imaging complex /VIC/;
- electromagnetic searcher /EMS/;
- multi-beam echo-sounder system /MB ESS/.

Let's consider operation peculiarities of these systems.

2.1 Video imaging complex for cable and thin pipeline recognition

Some communications (e.g. cables and pipelines with small diameter) can be detected on photo and video images as a pair of parallel lengthy lines. The basic function of VIC is data collection about inspected object and its identification (detection) on a seabed. Object identification consists of search of the longest straight line presented on the image. Hough transformation is used as the basic mathematical tool (Scherbatyuk, 1998). The recognition system output includes (fig. 1):

1. direction of the recognized object $\alpha_t^{<VIC>}$;
2. distance from video camera to the object $\rho_t^{<VIC>}$;
3. length of a visible part of the object $l_t^{<VIC>}$.

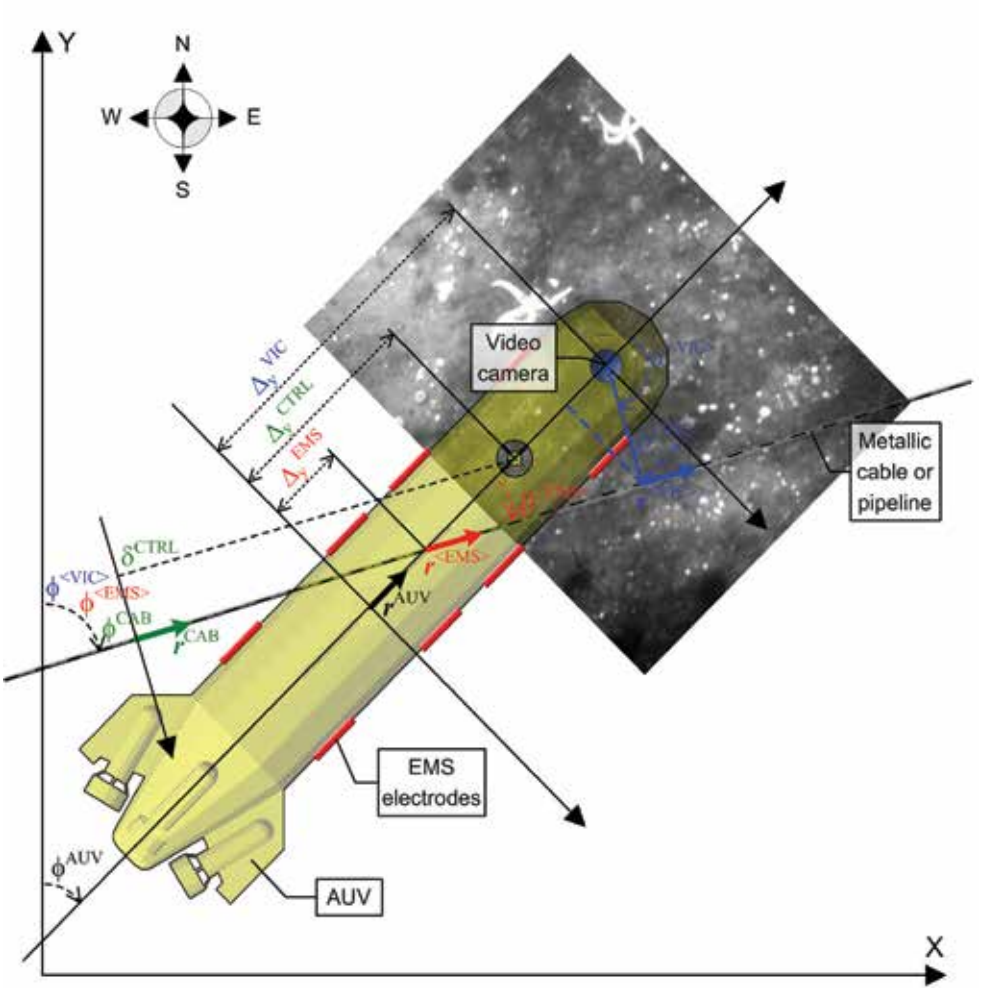


Fig. 1. AUV coordinate systems and devices arrangement

The position-direction vector $\boldsymbol{r}_{t<VIC>}=[X_{t<VIC>},Y_{t<VIC>},\Phi_{t<VIC>}]_T$ of recognized object in absolute coordinate system for a camera directed downwards perpendicularly to AUV shape is calculated as follows (neglecting AUV pitch and roll which during object inspection do not exceed several degrees):

$$\boldsymbol{r}_t^{<\mathrm{VIC}>} = \begin{bmatrix} \begin{bmatrix} X_t^{\mathrm{AUV}} \\ Y_t^{\mathrm{AUV}} \end{bmatrix} + \mathbf{G}(\Phi_t^{\mathrm{AUV}}) \cdot \left(\begin{bmatrix} \Delta_x^{\mathrm{VIC}} \\ \Delta_y^{\mathrm{VIC}} \end{bmatrix} + \begin{bmatrix} x'^{<\mathrm{VIC}>}_t \\ y'^{<\mathrm{VIC}>}_t \end{bmatrix} \right) \\ \Phi_t^{\mathrm{AUV}} + \alpha_t^{<\mathrm{VIC}>} - \mathrm{sign}\left(\alpha_t^{<\mathrm{VIC}>}\right) \cdot 90^0 \end{bmatrix} \tag{1}$$

$$x'^{<\mathrm{VIC}>}_t = \rho_t^{<\mathrm{VIC}>} \cdot \sin(\alpha_t^{<\mathrm{VIC}>}) \tag{2}$$

$$y'^{<\mathrm{VIC}>}_t = \rho_t^{<\mathrm{VIC}>} \cdot \cos(\alpha_t^{<\mathrm{VIC}>}) \tag{3}$$

$$\mathbf{G}(\varphi) = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{bmatrix} \tag{4}$$

Where: $x'_t{}^{<\mathrm{VIC}>}$, $y'_t{}^{<\mathrm{VIC}>}$ – object position on the video camera image; $\Delta_x{}^{\mathrm{VIC}}$, $\Delta_y{}^{\mathrm{VIC}}$ – onboard camera position; $\mathbf{G}(\varphi)$ – turn-matrix of coordinate system to an angle φ.

It is necessary to note that video recognition system does not guarantee reliable results due to noisy objects can be present on images (starfishes, alga, stones, sea garbage and so on) or inspected object can be invisible (buried or hidden by other objects). The probability $p_t{}^{<\mathrm{VIC}>}$ of correct recognition depends on identified length and direction of the line on the image. If line is long and lies down in appropriate direction then probability is high. Video imaging complex contacts and cable existence probability function during object inspection mission are shown on fig. 2.

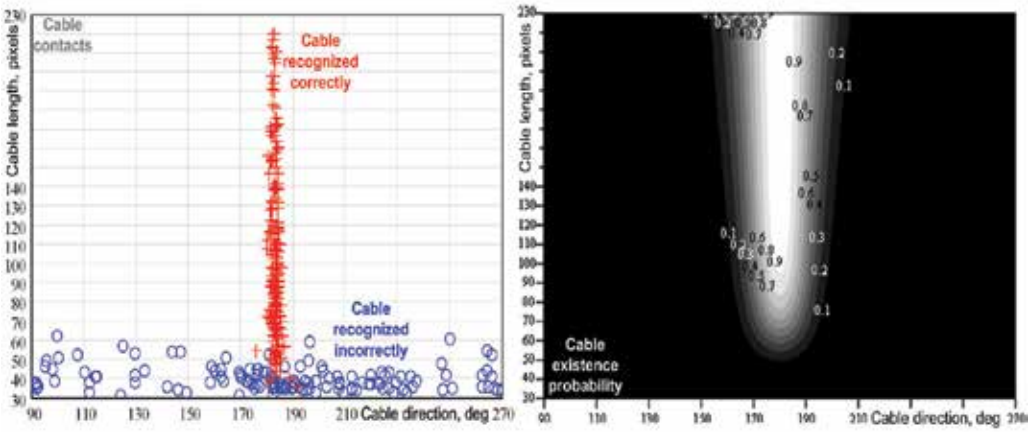


Fig. 2. VIC contacts filtration example: detected contacts (left image), existence probability (right image)

2.2 Electromagnetic searcher for metallic cable and pipeline recognition

Metallic communications (e.g. cables and pipelines) can be detected on the basis of electromagnetic methods.

EMS operation is based on electromagnetic field excitation in a water column and environment reaction measurement on this field (Kukarskih & Pavin, 2008). Electromagnetic Searcher (fig. 3) consists of three pairs of transmitting A_0, A_1, A_2 and receiving B_0, B_1, B_2 dipole antennas (TX- and RX- antennas) formed by corresponding electrodes $A_1'A_2'$-$A_1''A_2''$, A_1'-A_1'', A_2'-A_2'' and $B_1'B_2'$-$B_1''B_2''$, B_1'-B_1'', B_2'-B_2'' (electrodes of antennas A_0 and B_0 are paired).

TX-dipole antennas directed at a sharp angles α_0, α_1, α_2 with AUV longitudinal axis. Angles of RX-antennas (β_0, β_1, β_2) fitted so that transmitting electromagnetic field do not influence to corresponding RX-dipoles (mutual TX- and RX- antennas angles differ from 90° because of the field distortions by AUV shape). Such approach allows metallic object detection at any attack angle of the vehicle and the object.

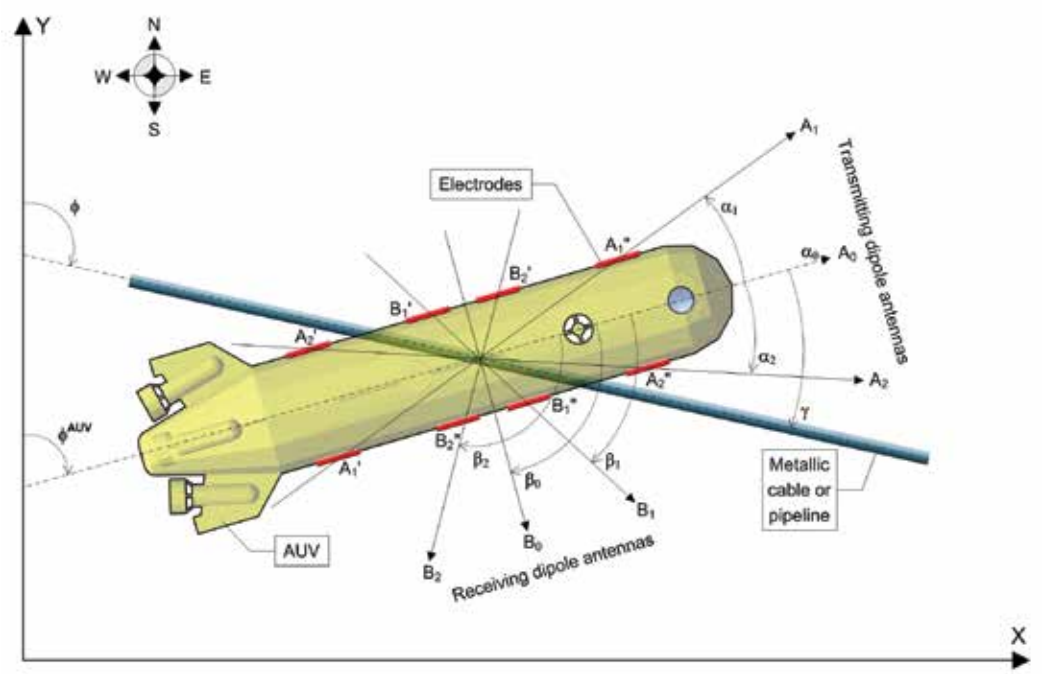


Fig. 3. Electrodes arrangement of EMS

Thus, alternately excite dipoles A_n (n=0,1,2) the current I_n will be induced in the object:

$$I_n = I^{A}_{n} \cdot R^{A}_{n}(\rho) \cdot \cos(\gamma - \alpha_n) \tag{5}$$

$$R^{A}_{n}(\rho) \approx K^{A}_{n} \cdot \rho^{-3} \tag{6}$$

Where: γ – angle between AUV longitudinal axis and lengthy metallic object; ρ – distance from the center of electrode system up to object; $R^{A}_{n}(\rho)$ – function of dependence of the current induced in a metallic object (by antenna A_n) from distance up to it; I^{A}_{n} – the current in dipole A_n; K^{A}_{n} – the constant coefficient dependent on AUV constructive properties, antenna A_n and electromagnetic properties of metallic object.

RX-antenna B_n measure potential difference U_n during transmitting of corresponding antenna A_n. Electrical field lines (induced in a metallic object) are directed along the lengthy object, because of this the potential difference U_n on antenna B_n depend on the following laws:

$$U_n = I_n \cdot R^{B}_{n}(\rho) \cdot \cos(\gamma - \beta_n) \tag{7}$$

$$R^{B}_{n}(\rho) \approx K^{B}_{n} \cdot \rho^{-2} \tag{8}$$

Where: $R^B_n(\rho)$ – potential difference function for antenna B_n (depend of distance from object); K^B_n – the constant coefficient characterizes antenna B_n.
Metallic object detection problem consist of calculation next two values: φ – object direction angle (fig. 3) and v – object vicinity to AUV (object existence probability assessment near the underwater vehicle). Value φ obtained by adding AUV heading (in absolute coordinate system) φ^{AUV} and AUV-object mutual angle γ: $\varphi = \varphi^{AUV} + \gamma$. Using (5) & (7) and replace $R_n(\rho) = I^A_n \cdot R^A_n(\rho) \cdot R^B_n(\rho)$ (n=0,1,2) the next system of equations can be obtained for synchronous detected potential difference U_n:

$$\begin{cases} U_n = R_n(\rho)\cdot\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n) \\ R_n(\rho) > 0 \\ n = 0,1,2 \end{cases} \tag{9}$$

Using equation (9) with n=1,2 and the fact that $R_1(\rho)\approx R_2(\rho)\approx R(\rho)$ (because A_1-B_1 and A_2-B_2 is symmetrical, see fig. 3) the next solving relatively γ can be obtained:

$$a\cdot\tan^2\gamma + b\cdot\tan\gamma + c = 0 \tag{10}$$

$$\gamma_{12} = \arctan\frac{-b\pm\sqrt{b^2-4\cdot a\cdot c}}{2\cdot a} \tag{11}$$

Where: $a = U_1\cdot(\sin\alpha_2\cdot\sin\beta_2) - U_2\cdot(\sin\alpha_1\cdot\sin\beta_1)$, $b = U_1\cdot(\sin\alpha_2\cdot\cos\beta_2 + \cos\alpha_2\cdot\sin\beta_2) - U_2\cdot(\sin\alpha_1\cdot\cos\beta_1 + \cos\alpha_1\cdot\sin\beta_1)$, $c = U_1\cdot(\cos\alpha_2\cdot\cos\beta_2) - U_2\cdot(\cos\alpha_1\cdot\cos\beta_1)$.
The next situations can be during calculating (11):

1. Equation (11) haven't real roots. It means that lengthy metallic object does not exist near the AUV (v=0).
2. Equation (11) have one or two real roots. The root $R_n(\rho)>0$ (n=0,1,2) will be chosen.

Object vicinity v can be estimated basing on $R_n(\rho)$ values calculated from (9). Each $R_n(\rho)$ (n=0,1,2) has maximum value when AUV situated over the metallic object. Besides, sensitivity of each transmitting-receiving electrode system A_n-B_n depends of angle γ (11). Using weighted sum of $R_n(\rho)$ with corresponding weights $|\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n)|$ for v calculation, the next equation is produced:

$$v = \frac{\sum_{n=0}^{2}\left|\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n)\right|\cdot R_n(\rho)}{\sum_{n=0}^{2}\left|\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n)\right|} \tag{12}$$

$$v = \frac{\sum_{n=0}^{2}U_n\cdot\operatorname{sign}\left(\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n)\right)}{\sum_{n=0}^{2}\left|\cos(\gamma-\alpha_n)\cdot\cos(\gamma-\beta_n)\right|} \tag{13}$$

EMS recognizes orientation of metallic lengthy object during mutually crossing movement. Metallic object position (on EMS data) is the position of electrodes system center at the moment of a potentials maximum on the receiving electrodes. Thus, vector of absolute coordinates may be calculated as follows:

$$\mathbf{r}_t^{<\mathrm{EMS}>} = \begin{bmatrix} \begin{bmatrix} X_t^{\mathrm{AUV}} \\ Y_t^{\mathrm{AUV}} \end{bmatrix} + \mathbf{G}(\Phi_t^{\mathrm{AUV}}) \cdot \begin{bmatrix} \Delta_x^{\mathrm{EMS}} \\ \Delta_y^{\mathrm{EMS}} \end{bmatrix} \\ \Phi_t^{\mathrm{AUV}} + \beta_t^{<\mathrm{EMS}>} \end{bmatrix} \tag{14}$$

Where: Δ_x^{EMS}, Δ_y^{EMS} – position of electrodes system center; $\beta_t^{<\mathrm{EMS}>}$ – mutual AUV-object angle (fig. 1).

EMS operates stably enough (identifies metallic lengthy object and define its crossing angle) at lengthy metallic object crossing under corners 30÷60 degrees to longitudinal AUV axis. The system operates uncertainly at AUV movement along the object (speed of change of electromagnetic potential is close to zero) and across (emitting electrodes direct too small potential). The estimation of lengthy metallic object existence probability $p_t^{<\mathrm{EMS}>}$ can be defined using electromagnetic field potential on the receiving electrodes $v_t^{<\mathrm{EMS}>}$ (a general potential estimation according to all receiving dipoles) and object direction $\Phi_t^{<\mathrm{EMS}>}$ of recognized object similarly to VIC contacts filtration (fig. 2).

2.3 Multi-beam echo-sounder system for pipeline recognition

It is also supposed, that for pipeline detection AUV uses the echosounder with relatively small number of beams (20-40) (fig. 4).

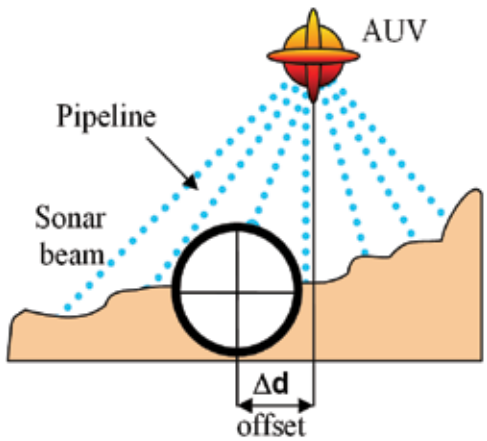


Fig. 4. AUV's sonar beam layout and data gathering process

Fuzzy logic methods were applied for identification of the pipeline (Pavin, 2006). The methods use the known pipeline parameters (diameter of its cross section and approximately known direction of its laying-out). The probability of pipeline correct identification depends on echo sounder and navigation data accuracy. These data compose AUV internal 3D terrain model. The main objective of identification system is an estimation of the pipeline presence probability in this or other place and determination of mutual position (offset) of AUV and the pipeline (fig. 4).

Offered method uses following information for the determination of pipeline exact position:

1. known seabed shape $M_{s,t}$ (it is created from each echo-sounder **s** at each moment t when data updating occurred) during time interval dt;
2. echo-sounder system accuracy (beam directional diagram angle δ_{ES} and distance accuracy Δ_{ES});

3. navigation system accuracy δ_{NAV};
4. known radius of the pipeline R_{PL};
5. φ_{PL}, ψ_{PL} – supposed direction and pipeline inclination;
6. the pipeline is a cylindrical object.

For identification of pipeline location it is enough to identify its top part ("ridge"). In this case the axis of the pipeline settles down below "ridge". The most probable location of pipeline axis is where maximal pipeline existence probabilistic assessment P and coordinates of axes are concentrated along some line. Thus, for determination of the pipeline location it is enough to present the received coordinates as points on the plane xOz (fig. 5) then it is possible to use fuzzy logic methods of lengthy objects determination based on point's position on the plane.

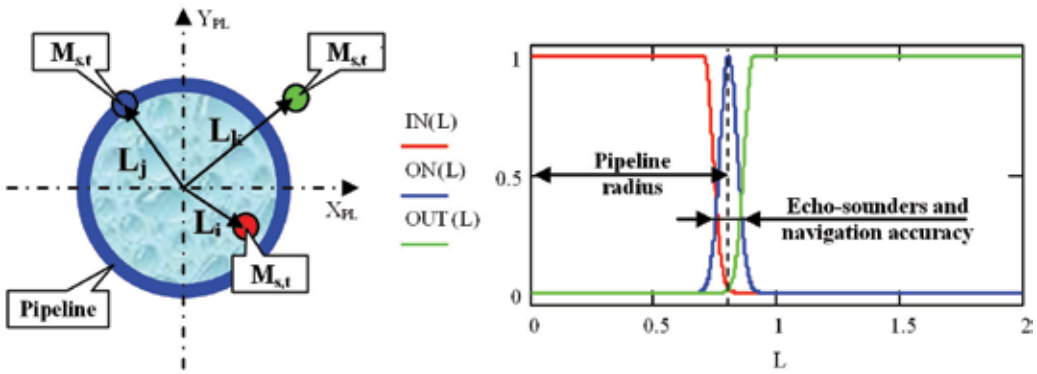


Fig. 5. Map points with pipeline projections (left image) and fuzzy logic values (right image)

The belonging probability of some point to the pipeline surface is appreciated according to following fuzzy rules:

1. The probability of the point belonging to the pipeline surface is higher, than more the neighbor points are "on a surface of the pipeline": *If L = "ON the pipe" then P := P + P_{true};*
2. If some point belongs to the pipeline surface then the neighbor points also lay on its surface, or are outside its borders. Otherwise the existence probability of the pipeline in the given point is reduced: *If L = "IN the pipe" then P := P – P_{false}.*

Values P_{true} and P_{false} take from modeling experiments, but supposed that $P_{true} < P_{false}$.. Fuzzy values *"IN the pipe"* and *"ON the pipe"* depend on echo-sounders accuracy, navigation accuracy and pipeline radius (fig. 5).

3. Recognition systems data integration

As was mentioned above, joint operation of various recognition systems is used for detection of objects of different classes. Multi-beam echo-sounders and electromagnetic searchers can be used for objects with the significant sizes of cross section (pipelines). Here EMS can be used only for detection of metallic object presence in AUV vicinity, but not for finding its orientation. EMS and video camera can be used for objects with the small area of cross section (cables and thin pipelines). Let's further consider procedure of data processing and object position finding for these two cases.

Data integration of Electromagnetic searcher and Video imaging complex are executed for reliable AUV movement during communication lines inspection. Probability $p_t^{<n>}$ is a factor of inspection object existence probability in time moment t regarding recognition system n. Accumulation of previous contacts can be made (fig. 6):

$$p_t^{<n>\Sigma} = k^{<n>} \cdot p_{t-1}^{<n>\Sigma} + p_t^{<n>} \tag{15}$$

Where: $p_t^{<n>\Sigma}$ – total size of contacts by the moment of time t for a subsystem n; $k^{<n>}$ – factor of attenuation ("forgetting" information of subsystem n).
The general parameter of object presence $p_t^{\Sigma\Sigma}$ according to all recognition systems is defined as the weighed sum of each subsystem estimations $p_t^{<n>\Sigma}$ and aprioristic estimation of object presence in the given point $w^0(X_t^{AUV}, Y_t^{AUV})$. Thus weights $w^{<n>}$ determine a degree of trust to each subsystem:

$$p_t^{\Sigma\Sigma} = \sum_n^N w^{<n>} \cdot p_t^{<n>\Sigma} + w^0\left(X_t^{AUV}, Y_t^{AUV}\right) \tag{16}$$

Average object direction and position are calculated on the basis of inspection object recognition systems data. The integrated estimation of cable or pipeline position and direction $\boldsymbol{r}_t^{CAB}=[X_t^{CAB}, Y_t^{CAB}, \Phi_t^{CAB}]^T$ (fig. 6) are similarly calculated, including data $\boldsymbol{r}_t^{<n>}=[X_t^{<n>}, Y_t^{<n>}, \Phi_t^{<n>}]^T$ of each subsystem n and aprioristic information about inspection object location in the given area $\boldsymbol{r}^0(X_t^{AUV}, Y_t^{AUV}) = [X^0(X_t^{AUV}, Y_t^{AUV}), Y^0(X_t^{AUV}, Y_t^{AUV}), \Phi^0(X_t^{AUV}, Y_t^{AUV})]^T$:

$$\boldsymbol{r}_t^{<n>\Sigma} = k^{<n>} \cdot \boldsymbol{r}_{t-1}^{<n>\Sigma} + p_t^{<n>} \cdot \boldsymbol{r}_t^{<n>} \tag{17}$$

$$\boldsymbol{r}_t^{CAB} = \frac{1}{p_t^{\Sigma\Sigma}} \cdot \left(\sum_{n=1}^{N} w^{<n>} \cdot \boldsymbol{r}_t^{<n>\Sigma} + w^0\left(X_t^{AUV}, Y_t^{AUV}\right) \cdot \boldsymbol{r}^0\left(X_t^{AUV}, Y_t^{AUV}\right) \right) \tag{18}$$

Where: $\boldsymbol{r}_t^{<n>\Sigma}=[X_t^{<n>\Sigma}, Y_t^{<n>\Sigma}, \Phi_t^{<n>\Sigma}]^T$ – temporary variable (a total component of each coordinate by the moment of time t for subsystem n).

4. AUV Control during cable or pipeline inspection

The AUV control during inspection phase supposes: a) time minimization of object search; b) data gathering about the object and surrounding environment (using video and echo-sounder information). Thus, it is possible to choose the following requirements to AUV movement organization:

1. During initial search stage it is necessary to use the object detection systems possessing the least probability of false operations and to move so that to maximize object detection probability.
2. After detection of any object it is necessary to make sure that the given object is required and proceed to its tracking.
3. AUV tracking trajectory should stretch above the object while there is a steady contact from identification systems.
4. When steady state contact is lost it is necessary to carry out a search trajectory (e.g. "divergent zigzag") to guarantee the return to the object.

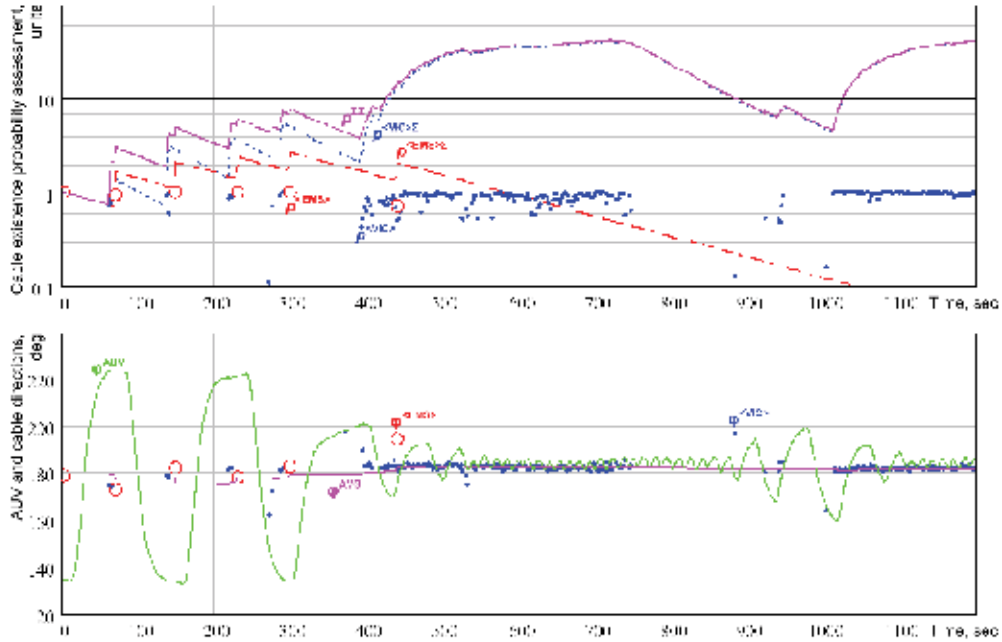


Fig. 6. Example of VIC and EMS data during cable inspection

4.1 Search the communication line

So, AUV behavior (Inzartsev & Pavin, 2006) during pipeline or cable search and detection is similar in many respects and consists of three steps (fig. 7).

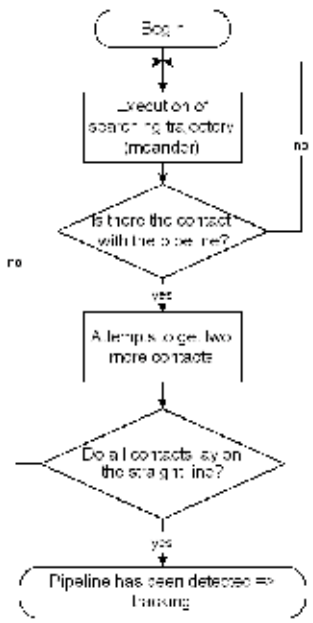


Fig. 7. Flowchart of preliminary phase of inspection

1. Execution of the search trajectory (meander) in prospective area of the object laying-out as long as crossing with the object looking like the lengthy.
2. Attempt to receive two more confirming contact from the object. For this purpose AUV changes motion direction on the opposite at each point of crossing (contact), i.e. goes by "zigzag". The primary goal of this maneuver is elimination of false contacts and calculation of approximate object direction (φ_{mean}).
3. If the number of crossing points is three, they lay on the straight line and the direction of this line coincides with the direction of the required object - then object is considered as found out and the stage of its tracking begins. Otherwise contacts are considered as false and the AUV returns to execution of the search trajectory (1).

4.2 AUV movement control during pipeline or cable tracking

AUV motion during the pipeline inspection has the features connected with freespan investigation. In the literature the task of main stage is not formalized and usually reduced to the list of the AUV desirable actions. In the paper the following sequence of actions is offered for the implementation of main stage (fig. 8):

1. AUV moves directly above the pipeline, thus:
   - leakage detection is fulfilled (the methane sensor is used usually for these purposes);
   - the video filming of the pipe surface is carried out;
   - SSS imaging is carried out on both sides of the pipeline for detection of extraneous objects;
   - pipe saggings and freespans are found out (on the basis of echo sounder data), and their lengths are estimated;
2. in the case of pipe sagging detection AUV moves away from the pipeline with backward motion for performance of SSS imaging;
3. AUV fulfills SSS imaging of sagging from both sides;
4. AUV comes back to the pipeline for continuation of inspection;
5. on completion of communication line inspection, the AUV fulfills backward motion for execution of SSS survey at the offset of 20-25 meters from inspected object.

AUV motion during cable tracking (Inzartsev & Pavin, 2008) is the same as for pipeline one (excluding item 2). The main goals of inspection can be achieved at AUV motion above the pipeline at height of 1-3 m. Thus, the inspection task is realized as the following algorithm (fig. 8).

The control heading Φ_t^{CTRL} during pipeline or cable tracking can be presented as the sum of the following values: direction of the cable/pipeline Φ_t^{CAB}, the crossing angle *cross*() of the inspection object (for "zigzag" trajectory) with corresponding sign $Side_t$ (determining a left-right-side AUV direction movement) and the trajectory stabilization function *stab*() above the inspection object (at movement lengthways the inspection object):

$$\Phi_t^{CTRL} = \Phi_t^{CAB} + Side_t \cdot cross\left(p_t^{\Sigma\Sigma}\right) - stab\left(\delta_t^{CTRL}\right) \tag{19}$$

$$Side_t = \begin{cases} -1, & \delta_t^{CTRL} > border\left(p_t^{\Sigma\Sigma}\right) \\ +1, & \delta_t^{CTRL} < -border\left(p_t^{\Sigma\Sigma}\right) \\ Side_{t-1}, & \left|\delta_t^{CTRL}\right| \le border\left(p_t^{\Sigma\Sigma}\right) \end{cases} \tag{20}$$

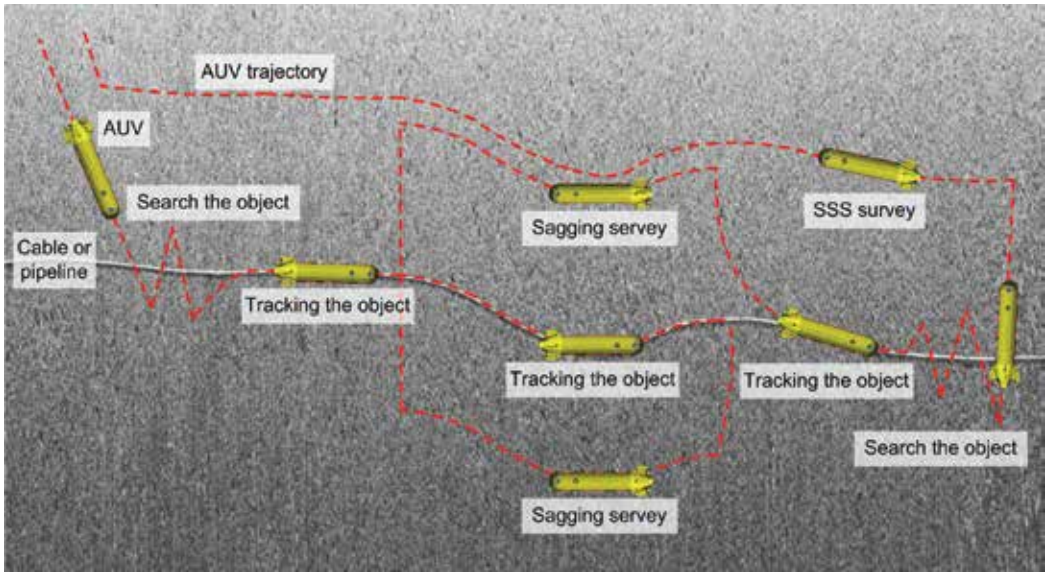


Fig. 8. AUV behavior scheme at inspection of the cable or pipeline

$$\delta_t^{\mathrm{CTRL}} = \left(X_t^{\mathrm{CTRL}} - X_t^{\mathrm{CAB}}\right)\cdot\cos\left(\Phi_t^{\mathrm{CAB}}\right) - \left(Y_t^{\mathrm{CTRL}} - Y_t^{\mathrm{CAB}}\right)\cdot\sin\left(\Phi_t^{\mathrm{CAB}}\right) \tag{21}$$

$$X_t^{\mathrm{CTRL}} = X_t^{\mathrm{AUV}} + \Delta_x^{\mathrm{CTRL}}\cdot\sin\left(\Phi_t^{\mathrm{AUV}}\right) \tag{22}$$

$$Y_t^{\mathrm{CTRL}} = Y_t^{\mathrm{AUV}} + \Delta_y^{\mathrm{CTRL}}\cdot\cos\left(\Phi_t^{\mathrm{AUV}}\right) \tag{23}$$

Where: δ_t^{CTRL} – calculated distance from cable/pipeline to "AUV stabilization point"; *border*() – size of the inspection border zone. The "AUV stabilization point" is located on the vehicle shape with coordinates: Δ_x^{CTRL}, Δ_y^{CTRL} – in the connected coordinate system and X_t^{CTRL}, Y_t^{CTRL} – in absolute coordinate system.

AUV search trajectory is characterized by the crossing angle function *cross*() and the trajectory stabilization function *border*(). Examples of the possible dependences used during AUV field tests are given below:

$$border\left(p^{\Sigma\Sigma}\right) = \delta_{min}^{border} + \delta_{max}^{border}\cdot\exp\left(-K^{border}\cdot p^{\Sigma\Sigma}\right) \tag{24}$$

$$cross\left(p^{\Sigma\Sigma}\right) = \varphi_{min}^{cross} + \varphi_{max}^{cross}\cdot\exp\left(-K^{cross}\cdot p^{\Sigma\Sigma}\right) \tag{25}$$

$$stab\left(\delta^{\mathrm{CTRL}}\right) = \varphi_{max}^{stab}\cdot\tanh\left(K^{stab}\cdot\delta^{\mathrm{CTRL}}\right) \tag{26}$$

Where: δ_{min}^{border} и δ_{max}^{border} – minimal and maximal border size of the search zone; K^{border} – coefficient of border reduction; φ_{min}^{cross} и φ_{max}^{cross} – minimal and maximal cross angle; K^{cross} – coefficient of the cross angle reduction; φ_{max}^{stab} – maximal angle of mutual AUV-to-object offset compensation; K^{stab} – AUV trajectory stabilization coefficient at movement lengthways the inspection object.

AUV control system calculates object survey zone and control heading. Thus, the trajectory of underwater vehicle will represent oscillatory movement along the inspection object (fig.

6, 13). The amplitude of fluctuations is inversely to the object existence probability near the AUV. The maximal estimation of the probability changes movement to a direct line. When estimation is reduced (loss of the object) the oscillatory movements appear again.

5. Modeling experiments and sea trial results

The proposed approach consistency was proved by means of modeling experiments (for the ESS-data using) and by sea trials (for jointly using of VIC and EMS-data).

5.1 Modeling experiments of pipeline inspection

Investigation of AUV behavior during inspection of pipeline, which is partly filled up with ground, was carried out by means of modeling. The modeling environment includes:

- 3D-model of bottom surface and artificial objects (pipeline in this case);
- AUV dynamic and kinematics models;
- echo-sounder model.

The model of bottom surface (fig. 9) along with model of artificial objects allows to form any terrain shapes, including concave and ambiguous (Pavin, 2005). The model includes three basic levels of detailed elaboration. *The micro* level corresponds to accuracy of the echo sounder (centimeters). *The middle* level corresponds to AUV motion during updating the data (decimeters). *The macro* level describes the general bottom characteristics. The model allows to set characteristics for each of three levels. For more careful test of identification algorithms the terrain model has been added with an opportunity of generation: "visor" and "stone". The macro-relief is used as a skeleton on which middle- and micro-levels are placed according to the certain rules.

The echo sounder model is applicable for calculation as single beam (with the fixed diagram of an orientation), and multi-beam echo-sounders. Thus, the basic characteristics of the echo sounder are taken into account: their layout onboard the AUV and the sensor direction, range and precision of measurements, directional diagram.

The echo sounder was modeled by several acoustic beams (from 5 up to 20) directed fan-shaped downwards in sector of 40°. AUV mission consisted of detection of the pipeline, motion along it, pipeline sagging detection and its contouring. Many variants of terrain, pipeline position and various number of echo sounder beams were modeled.

In fig. 9-11 the results of partly buried pipeline detection are demonstrated. The pipeline diameter is 1.6 m. The level of pipeline hiding in sea bottom is 50% (on average). The number of used echo sounder beams - 10. Motion altitude is 3 m. Motion speed is about 0.8 m/s and depends on echo sounder data update rate.

Scheme of modeling experiment is presented on fig. 9. Terrain “coverage” by echo sounders is represented by blue strip of variable width along the AUV lane (left picture). Right picture illustrates 3D-view of the main inspection phase. It includes freespan detection and survey. AUV has moved along the pipeline, found out sagging, contoured it and renewed motion further along the pipeline. In fig. 10 echo sounder data of AUV portside and starboard are shown (5 sonar beams from each side). The figure presents results of automatic sagging detection on the base of these data. In fig. 11 the probability of pipeline presence in echo sounder field of vision is illustrated. It is supposed that pipeline “exists” if this value is positive. In this case AUV offset is calculated (blue points on the graph) and used in control.

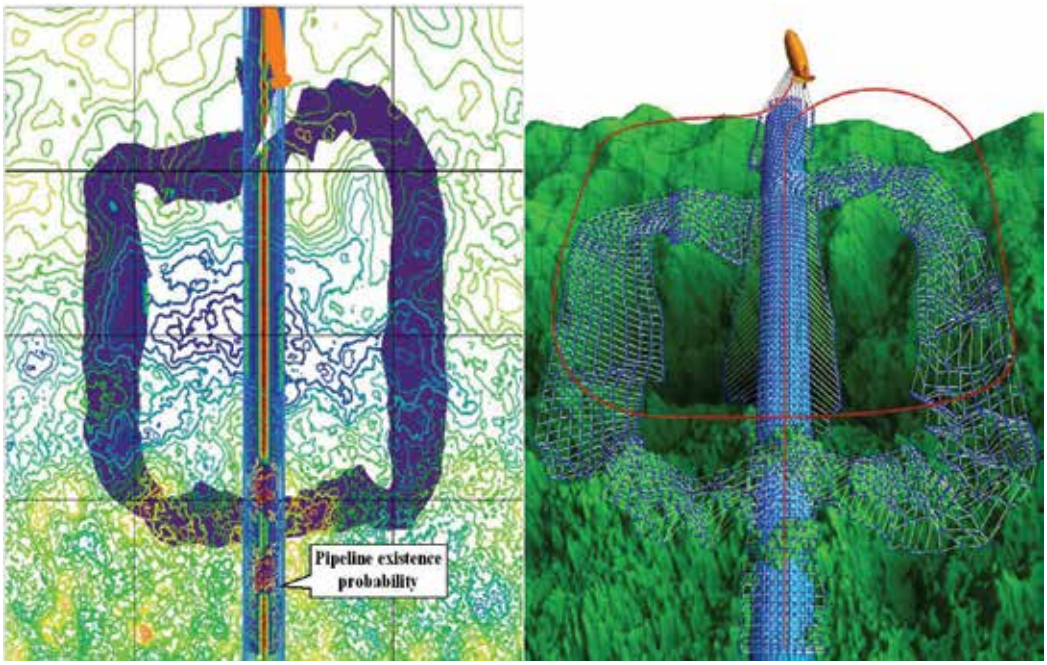


Fig. 9. Pipeline detection (left picture) and 3D-view of pipeline inspection (right picture)

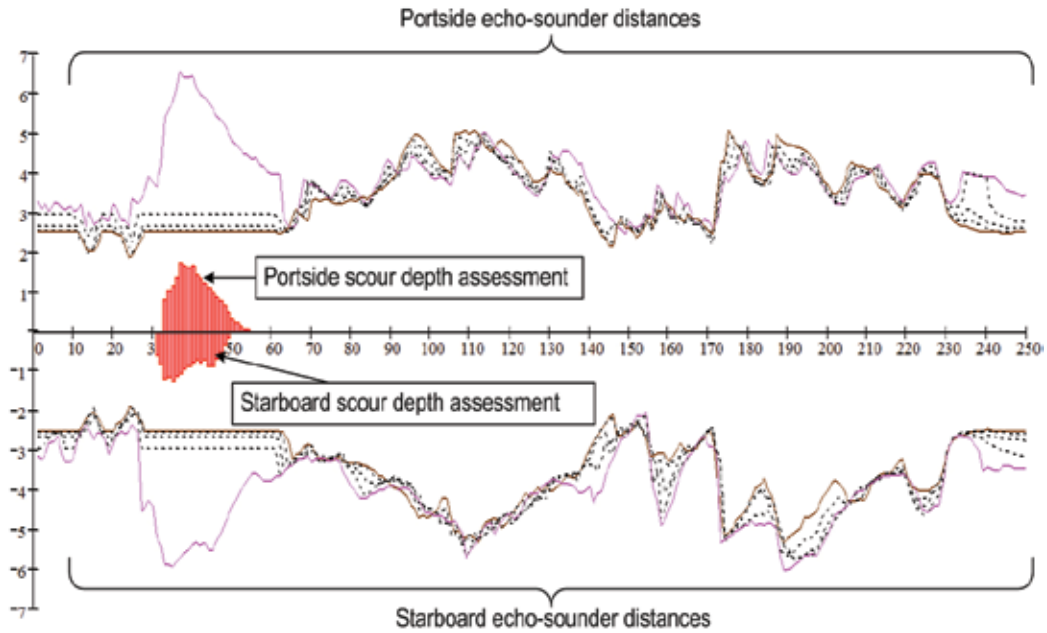


Fig. 10. Echo sounder data and pipeline freespan assessment during sagging inspection

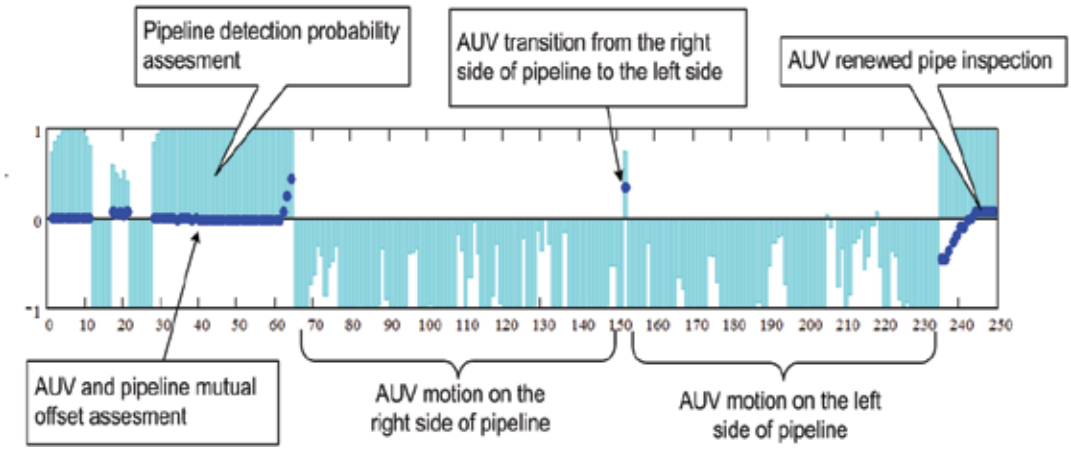


Fig. 11. Pipeline detection probability and mutual offset assessment

5.2 Sea trial results of cable inspection

The first full-scale experiments were made with the use of VIC and EMS developed in the Institute of Marine Technology Problems. Both devices were installed aboard AUV MT-98 (Fig. 12) and used for automatic inspection of thin metallic cable, which was preliminary laid on the bay floor.

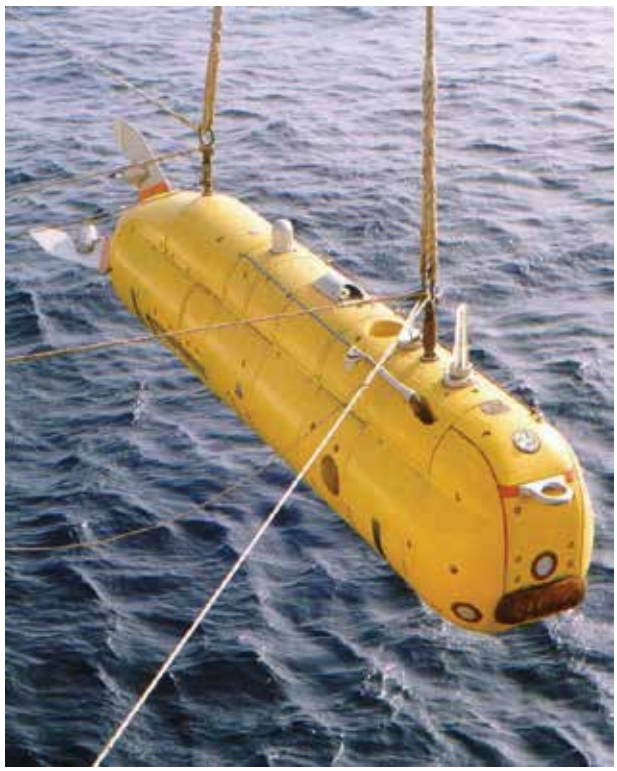


Fig. 12. AUV MT-98

The described method was realized in a coordinating level of AUV control system. AUV sea trials were carried out in two stages (fig. 13):

1. EMS only was involved as a cable recognition system,,
2. EMS and VIC were used for cable detection simultaneously.

Sea trial parameters and some results are summarized in table 1.

Parameter	Value
Cable diameter	12 mm
Cable length	800 m
Approx cable direction	180°
Average AUV altitude	1.7 m
Average AUV velocity	0.35 m/s
Average inspection speed (EMS only detection using)	0.2 m/s
Average inspection speed (EMS & VIC detection using)	0.3 m/s
AUV-cable offset dispersion (EMS only detection using)	1.5 m
AUV-cable offset dispersion (EMS & VIC detection using)	0.5 m
Maximal AUV-cable offset (EMS only detection using)	3 m
Maximal AUV-cable offset (EMS & VIC detection using)	1.5 m
Cable presence on photo images (EMS only detection using)	30%
Cable presence on photo images (EMS & VIC detection using)	90%

Table 1. Field Test Parameters and Results

In both cases after first EMS-contact underwater vehicle made “zigzag” movement along a general cable direction to collect the contact statistic. As four minutes passed (from the first EMS contact) AUV started search movements guided by inspection zone borders. In the result all test part of the cable was inspected by AUV in autonomous mode (fig. 13, 14-left picture).

The second full-scale experiment was carried out during the inspection of real cable, which has been laid several years ago on bottom of Ussuriysky Bay of Japan Sea. The cable is partly silted and also surrounded by sea organisms (fig. 14 central pictures).

AUV moved above silted cable fragments by "zigzag" and straight forward along well seen sites of cable. The total length of the surveyed fragment of the cable composed 13 km. The scheme of inspection consisted of two stages. At the first stage the AUV moved at height of 1.7 m with the use of EMS & VIC, specified cable position and carried out its photographing. The control system registered coordinates of intersection points with the cable. At the second stage the AUV rose on height of 5 m, departed aside from the cable approximately on 20-25 m and came back along the cable in an initial point (registered coordinates were used). At the same time HF SSS (fig. 14 right picture) survey of cable vicinities was being carried out (i.e. the motion was similar to inspection of pipeline freespan).

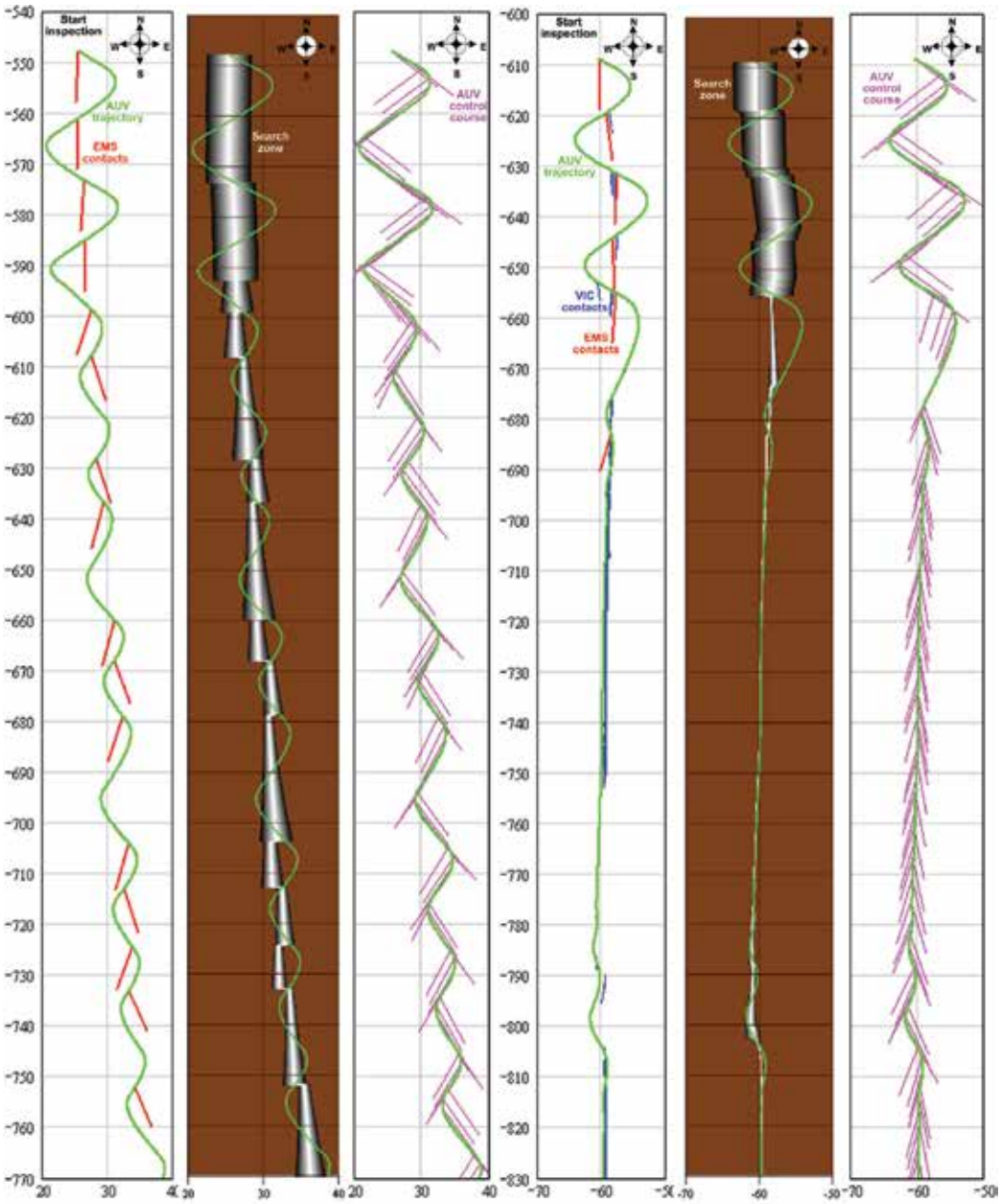


Fig. 13. AUV cable inspection with using of EMS only (left) and using EMS & VIC (right)

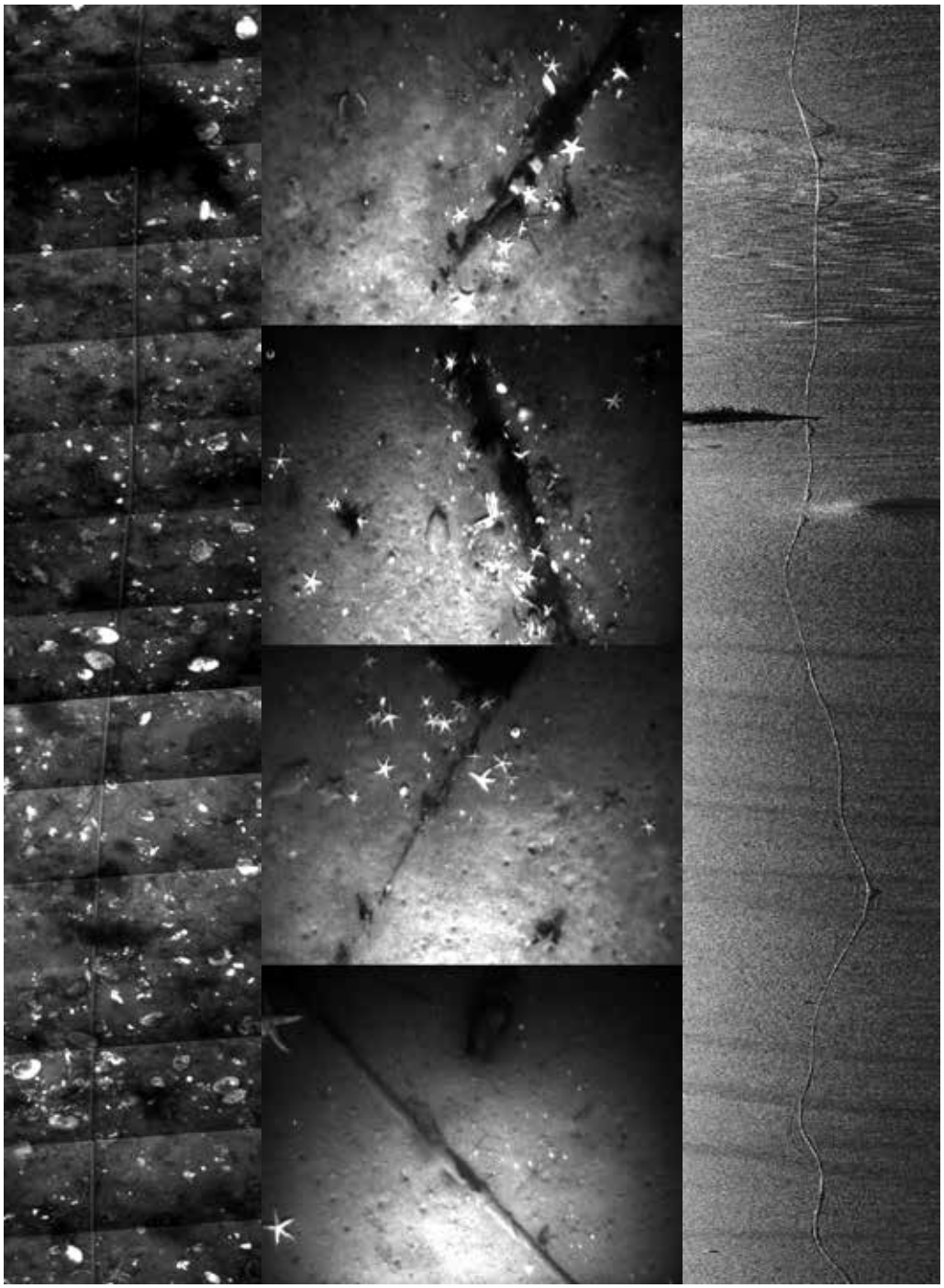


Fig. 14. Photo-images sequences during AUV movement along a cable (left picture), photo-images (central pictures) and HF-SSS image (right picture) of partly visible cable

6. Conclusion

Described method of data calculation and AUV control model are used in underwater vehicles designed in Institute of Marine Technology Problems. Sea trial results confirmed high performance of the VIC & EMS integration method. The modeling results of ESS allow to appreciate optimistically an opportunity of the algorithm using for solving of pipeline inspection problems. The method has shown the serviceability and to a first approximation it is supposed to use in the control system of existing AUV for detection and the motion organization along the acoustically visible lengthy cylindrical objects (in particular, pipelines). However the method requires testing with use of real ESS and the pipeline.
Future method development consists of data integration from low frequency & high frequency SSS for pipeline and cable tracking. Experiments with the use of Hough transformation for detection of lengthy objects on SSS-images yield encouraging results.
Authors thank IMTP directions for R&D support and colleagues – for the help in carrying out of sea tests.

7. References

Ageev, M. (2005). AUV Equipment and Control While Conducting Investigation of Underwater Pipeline (in Russian), *Underwater Technologies*, №1, pp. 68-72, Dalnauka, Vladivostok, Russia

Asai, T.; Kojima, J.; Asakawa, K. & Iso, T. (2000). Inspection of submarine cable of over 400 km by AUV, *Proceedings of the 2000 International Symposium on Underwater Technology, UT 00*, pp. 133 – 135, May 2000, Tokyo, Japan

Baker, J. (1991). Alternative Approaches to Pipeline Survey. *Subtech'91,* Vol. 27, pp. 333-345.

Evans, J.; Petillot, Y.; Redmond, P.; Wilson, M. & Lane, D. (2003). AUTOTRACKER: Real-Time Architecture for Pipeline and Cable Tracking on AUVs, *Proceedings of 13th International Symposium On Unmanned Untethered Submersible Technology (UUST03)*

Inzartsev, A. & Pavin, A. (2006). AUV Behavior Algorithm While Inspecting of Partly Visible Pipeline, *Proceedings of the OCEANS 2006 MTS/IEEE Conference*, ISBN 1-4244-0115-1, September 2006, Boston, MA, USA

Inzartsev, A. & Pavin, A. (2008). AUV Cable Tracking System Based on Electromagnetic and Video Data, *Proceedings of OCEANS'08 MTS/IEEE Kobe-Techno-Ocean'08 (OTO'08)*, ISBN: 978-1-4244-2126-8, April 2008, Kobe, Japan

Kogan, I.; Paull, C. at al. (2006). ATOC/Pioneer Seamount cable after 8 years on the seafloor: Observations, environmental impact, *Continental Shelf Research*, Volume 26, Issue 6, April 2006, pp. 771-787

Kojima, J.; Ito, Y.; Asakawa, K. et al. (1997). Development of Autonomous Underwater Vehicle "Aqua Exporer 2" for Inspection of Underwater Cables. *Proc. of MTS/IEEE, Oceans'97,* Canada

Kukarskih, A. & Pavin, A. (2008). Using of Electromagnetic Searcher for Inspection of Metallic Cable by Means of AUV, *Proceedings of OCEANS'08 MTS/IEEE Kobe-Techno-Ocean'08 (OTO'08)*, ISBN: 978-1-4244-2126-8, April 2008, Kobe, Japan

Matsumoto, S. & Ito, Y. (1995), Real-Time Vision-Based Tracking of Submarine-Cables for AUV/ROV, *Proc. of Conf. Oceans'95*

Murray, B. (1991). Pipeline Freespan Monitoring. *Subtech'91,* Vol. 27, pp. 347-354

Ortiz, A.; Simó, M. & Oliver, G. (2000). Image Sequence Analysis for Real-Time Underwater Cable Tracking, *Fifth IEEE Workshop on Applications of Computer Vision (WACV'00)*, p. 230

Pavin, A. (2005). Simulation Environment for AUV Control Algorithms Development, *Proceedings of 14th International Symposium on Unmanned Untethered Submersible Technology*, August 2005, Durham, New Hampshire, USA

Pavin, A. (2006). The Pipeline Identification Method Basing on AUV's Echo-Sounder Data, *Proceedings of the OCEANS 2006 MTS/IEEE Conference*, ISBN 1-4244-0115-1, September 2006, Boston, MA USA

Petillot, Y.; Reed, S. & Bell, J. (2002). Real time AUV pipeline detection and tracking using side scan sonar and multi-beam echosounder, *Oceans '02 MTS/IEEE*, Volume: 1, pp. 217-222, October 2002

Scherbatyuk, A. (1998). Comparison of Methods for Identifying Objects with Rectilinear Edges on Underwater Video Images. *Pattern Recognition and Image Analysis*, Vol. 8, № 3, 1998, pp. 467 - 469

Scherbatyuk, A.; Boreyko, A. & Vaulin, Yu. (2000). AUV Operation Based on Video Data Processing: Some IMTP Experience, *Workshop on Sensors and Sensing Technology for Autonomous Ocean Systems,* Oct -Nov 2000, Hawaii

13

Thruster Modeling and Controller Design for Unmanned Underwater Vehicles (UUVs)

Jinhyun Kim
Seoul National University of Technology
Republic of Korea

1. Introduction

Thruster modeling and control is the core of underwater vehicle control and simulation, because it is the lowest control loop of the system; hence, the system would benefit from accurate and practical modeling of the thrusters. In unmanned underwater vehicles, thrusters are generally propellers driven by electrical motors. Therefore, thrust force is simultaneously affected by motor model, propeller map, and hydrodynamic effects, and besides, there are many other facts to consider (Manen & Ossanen, 1988), which make the modeling procedure difficult. To resolve the difficulties, many thruster models have been proposed.

In the classical analysis of thrust force under steady-state bollard pull conditions, a propeller's steady-state axial thrust (T) is modeled proportionally to the signed square of propeller shaft velocity (Ω),$T=c_1\Omega|\Omega|$ (Newman, 1977). Yoerger et al. (Yoerger et al., 1990) presented a one-state model which also contains motor dynamics. To represent the four-quadrant dynamic response of thrusters, Healey et al. (Healey et al., 1995) developed a two-state model with thin-foil propeller hydrodynamics using sinusoidal lift and drag functions. This model also contains the ambient flow velocity effect, but it was not dealt with thoroughly. In Whitcomb and Yoerger's works (Whitcomb & Yoerger, 1999a; Whitcomb & Yoerger, 1999b), the authors executed an experimental verification and comparison study with previous models, and proposed a model based thrust controller. In the two-state model, lift and drag were considered as sinusoidal functions, however, to increase model match with experimental results, Bachmayer et al. (Bachmayer et al., 2000) changed it to look-up table based non-sinusoidal functions, and presented a lift and drag parameter adaptation algorithm (Bachmayer & Whitcomb, 2003). Blanke et al. (Blanke et al., 2000) proposed a three-state model which also contains vehicle dynamics. Vehicle velocity effect was analyzed using non-dimensional propeller parameters, thrust coefficient and advance ratio. However, in the whole range of the advance ratio, the model does not match experimental results well.

In the former studies, there are three major restrictions. First, thruster dynamics are mostly modeled under the bollard pull condition, which means the effects of vehicle velocity or ambient flow velocity are not considered. However, while the thruster is operating, naturally, the underwater vehicle system is continuously moving or hovering against the current. In addition, the thrust force would be degraded by up to 30% of bollard output due to ambient flow velocity. Therefore, the bollard pull test results are only valid at the

beginning of the operation, and the ambient flow velocity induced by vehicle movement or current must be taken into consideration. Moreover, non-parallel ambient flow effects have received less attention in previous works (Saunders & Nahon, 2002). These are dominant when an underwater vehicle changes its direction, or when an omni-directional underwater vehicle with non-parallel thrusters like ODIN (Choi et al., 1995) is used. Non-parallel ambient flow effects could be modeled simply by multiplying the ambient flow by the cosine function, but experimental results have been inconsistent. Second, in the models including the ambient flow effect, the thrust equations are derived from approximations of empirical results without concern for physical and hydrodynamic analysis. This leads to a lack of consistency in the whole thrust force map, especially, when the directions of thrust force and ambient flow velocity are opposite. Third, most of the previous models contain axial flow velocity of the thruster, because the models are usually based on Bernoulli's equation and momentum conservation. However, measuring axial flow velocity is not feasible in real systems, so we cannot apply those equations directly to the controller. Hence, in Fossen and Blanke's work (Fossen & Blanke, 2000), the authors used an observer and estimator for the axial flow velocity. And, Whitcomb and Yoerger (Whitcomb & Yoerger, 1999b) used the desired axial velocity as an actual axial flow velocity for the thrust controller. Those approaches, however, increase the complexity of controller.

To resolve the above restrictions, in this article, we mainly focus on steady-state response of thrust force considering the effects of ambient flow and its incoming angle, and propose a new thruster model which has three outstanding features that distinguish it from other thruster models. First, we define the axial flow velocity as the linear combination of ambient flow velocity and propeller shaft velocity, which enables us to precisely fit the experimental results with theoretical ones. The definition of axial flow gives a physical relationship between the momentum equation and the non-dimensional representation, which has been widely used to express the relation between ambient flow velocity, propeller shaft velocity, and thrust force. Also, the modeling requires only measurable states, so it is practically feasible. Second, we divide the whole thrust force map into three states according to the advance ratio. The three states, equi-, anti-, and vague directional states, explain the discontinuities of the thrust coefficient in the non-dimensional plot. While the former approaches failed to consider anti- and vague directional states, the proposed model includes all of the flow states. Here, we define the value of border status between anti- and vague directional states as *Critical Advance Ratio (CAR)* where the patterns of streamline change sharply. The details will be given in Section 3. Third, based on the two above features, we develop the incoming angle effects to thrust force. Incoming angle means the angle between ambient flow and thruster, which is easily calculated from vehicle velocity. If the incoming angle is 0 degree, the thrust force coincides with the equi-directional state, or if the angle is 180 degree, the thrust force coincides with the vague or anti-directional state according to the advance ratio. It should be pointed out that the mid-range of incoming angle cannot be described by a simple trigonometric function of advance ratio. So we analyze the characteristics of incoming angle, and divide the whole angle region into the three states above. Also, for the border status among the states, *Critical Incoming Angle (CIA)* is defined.

This chapter is organized as follow: In Section 2, the thruster modeling procedure will be explained and a new model for the thruster is derived. Section 3 addresses three fluid states

with CAR and CIA, and explains the physical meanings. Then Section 4 describes the matching results of experiments with the proposed model simulation, and compares these results with conventional thrust models. Section 5 describes the thruster controller based on the proposed model. Finally, concluding remarks will summarize the results.

2. Basic thruster dynamics model

2.1 Thruster dynamic based on axial flow velocity

The propeller is represented by an actuator disk which creates across the propeller plane a pressure discontinuity of area A_p and axial flow velocity u_p. The pressure drops to p_a just before the disk and rises to p_b just after and returns to free-stream pressure, p_∞, in the far wake. To hold the propeller rigid when it is extracting energy from the fluid, there must be a leftward thrust force T on its support, as shown in Fig. 1.

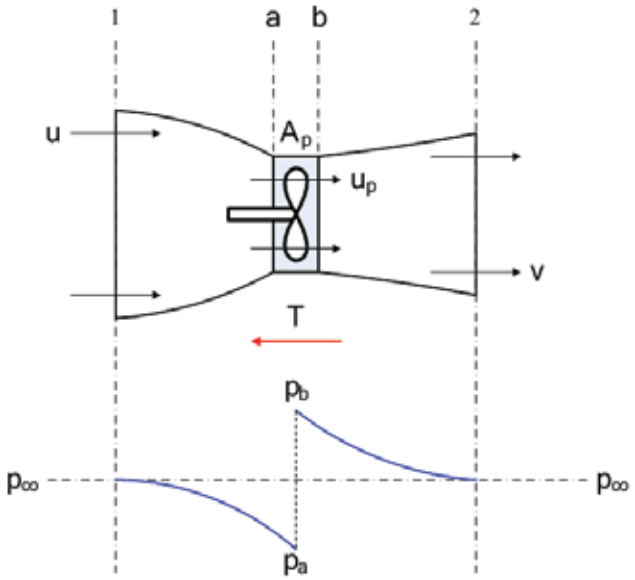


Fig. 1. Propeller race contraction; velocity and pressure changes

If we use the control-volume-horizontal-momentum relation between sections 1 and 2,

$$T = \dot{m}(v - u). \tag{1}$$

A similar relation for a control volume just before and after the disk gives

$$T = A_p(p_b - p_a). \tag{2}$$

Equating these two yields the propeller force

$$T = A_p(p_b - p_a) = \dot{m}(v - u). \tag{3}$$

Assuming ideal flow, the pressures can be found by applying the incompressible Bernoulli relation up to the disk

$$\begin{aligned} \textit{From 1 to a:} \quad & p_\infty + \frac{1}{2}\rho u^2 = p_a + \frac{1}{2}\rho u_p^2, \\ \textit{From b to 2:} \quad & p_\infty + \frac{1}{2}\rho v^2 = p_b + \frac{1}{2}\rho u_p^2. \end{aligned} \tag{4}$$

Subtracting these and noting that $\dot{m} = \rho A_p u_p$ through the propeller, we can substitute for p_b - p_a in Eq. (3) to obtain

$$p_b - p_a = \frac{1}{2}\rho\left(v^2 - u^2\right), \tag{5}$$

or

$$u_p = \frac{1}{2}\left(v + u\right) \Rightarrow v = 2u_p - u. \tag{6}$$

Finally, the thrust force by the disk can be written in terms of u_p and u by combining Eqs. (5) and (6) as follows:

$$T = 2\rho A_p u_p (u_p - u). \tag{7}$$

Up to this point, the procedures are the same as the previous approaches. Now, we define the axial flow velocity as

$$u_p \triangleq k_1 u + k_2 D\Omega, \tag{8}$$

where k_1 and k_2 are constant. The schematic diagram of the axial flow relation is shown in Fig. 2.

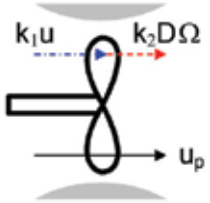


Fig. 2. Proposed axial flow model

For quasi-stationary flow, the axial flow only depends on ambient flow and propeller rotational motion. More complex combinations of ambient flow and propeller velocity are possible, but this linear combination is adequate as will be shown later. This somewhat simplified definition gives lots of advantages and physical meanings.
Finally, substituting Eq. (8) to Eq. (7), the proposed thrust model can be derived as follows:

$$\begin{aligned} T &= 2\rho A_p (k_1 u + k_2 D\Omega)(k_1 u + k_2 D\Omega - u), \\ &= 2\rho A_p (k_1' u^2 + k_2' u D\Omega + k_3' D^2\Omega^2). \end{aligned} \tag{9}$$

This model will be used in the following non-dimensional analysis.

2.2 Non-dimensional analysis

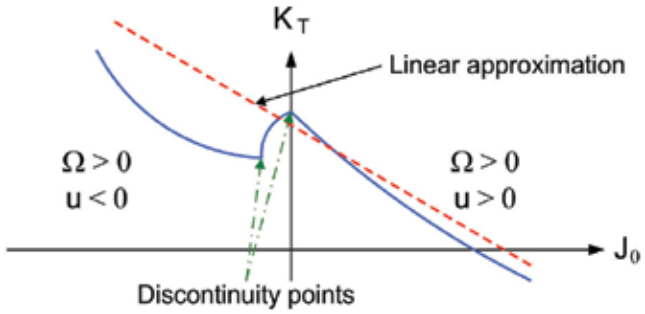


Fig. 3. Thrust coefficient as a function of advance ratio and its linear approximation

The non-dimensional representation for thrust coefficient has been widely used to express the relation between thrust force, propeller shaft velocity and ambient flow velocity as below:

$$K_T(J_0) = \frac{T}{\rho D\Omega|\Omega|}, \tag{10}$$

where $J_0 = u/D\Omega$ is the advance ratio. Figure 3 shows a typical non-dimensional plot found in various references (Manen and Ossanen, 1988; Blanke et al., 2000). In former studies, the non-dimensional relation is only given as an empirical look-up table or simple linear relationship for the whole non-dimensional map as (Fossen & Blanke, 2000).

$$K_T(J_0) = a_1 J_0 + a_2. \tag{11}$$

However, as shown in Fig. 3, Eq. (11) cannot accurately describe the characteristics of the thrust coefficient, especially when $J_0<0$, and, rather than a linear equation, the thrust coefficient seems to be close to a quadratic equation except for the discontinuity points. Even more, Eq. (11) has no physical relationship with thrust force, but is just a linear approximation from the figure.

The proposed axial flow assumption would give a solution for this. The non-dimensionalization of Eq. (9) is expressed as

$$\frac{T}{\rho D^4\Omega^2} = \frac{\pi}{2}\left[k_1'\left(\frac{u}{D\Omega}\right)^2 + k_2'\left(\frac{u}{D\Omega}\right) + k_3'\right]. \tag{12}$$

And, the quadratic thrust coefficient relation is obtained as following:

$$K_T(J_0) = \frac{\pi}{2}\left[k_1' J_0^{\,2} + k_2' J_0 + k_3'\right]. \tag{13}$$

Hence, contrary to other models, the axial flow definition of Eq. (8) gives an appropriate relationship between the thrust force equation and non-dimensional plot since the derivation was done by physical laws. Also, Eq. (13) can explain the characteristics of the

quadratic equation of the thrust coefficient. From this phenomenon, we can perceive that the axial flow definition in Eq. (8) is reasonable. The coefficients of quadratic equations could be changed depending on hardware characteristics. However, there is still a question of the discontinuities of thrust coefficient in Fig. 3 which has not been answered yet by existing models. This problem will be addressed in the following section.

3. Thrust force with ambient flow model

If ambient flow varies, thrust force changes even with the same propeller shaft velocity, which means that the ambient flow disturbs the flow state under the bollard pull condition. Flow state is determined by a complex relation between propeller shaft velocity, ambient flow velocity and its incoming angle. This will be shown in the following subsections.

3.1 Flow state classification using CAR

In this subsection, we define three different flow states according to the value of advance ratio and the condition of axial flow. To distinguish them, we introduce *Critical Advance Ratio (CAR), J^**.

The three states are as below:

- Equi-directional state

$$J_0 > 0, \tag{14}$$

$$u_p = k_1 u + k_2 D\Omega > 0. \tag{15}$$

- Anti-directional state

$$J^* < J_0 < 0, \tag{16}$$

$$u_p = k_1 u + k_2 D\Omega > 0. \tag{17}$$

- Vague directional state

$$J^* > J_0, \tag{18}$$

$$u_p = k_1 u + k_2 D\Omega < 0. \tag{19}$$

Figure 4 shows the flow states schematically.

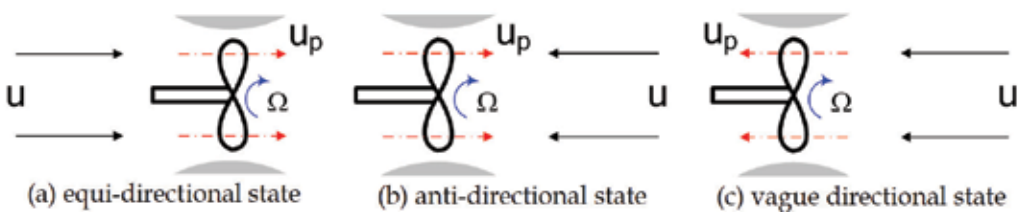


Fig. 4. Three flow states

Equi-directional state: The equi-directional state occurs when the ambient flow direction and axial flow direction coincide. In this state, if the ambient flow velocity increases, the pressure difference decreases. Hence the thrust force reduces, and the streamline evolves as a general form. (Fig. 4(a))
Anti-directional state: The anti-directional state happens when the ambient flow and axial flow direction are opposite. However, the axial flow can thrust out the ambient flow, hence the streamline can be built as sink and source. The Bernoulli equation can be applied and the thrust equation is still valid but the coefficients are different from those of the equi-directional state. Also, the thrust force rises as the ambient flow velocity increases, because the pressure difference increases. (Fig. 4(b))
Vague directional state: In the vague directional state, the axial flow cannot be well defined. The axial flow velocity cannot thrust out the ambient flow; hence the direction of axial flow is not obvious. This ambiguous motion disturbs the flow, so the thrust force reduces. In this case, we cannot guess the form of the streamline, so the thrust relation cannot be applied. However, the experimental results show the proposed thrust relation is still valid in this state. (Fig. 4(c))

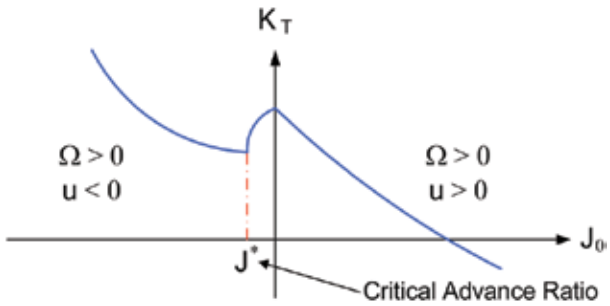


Fig. 5. Thrust coefficient as a function of advance ratio and Critical Advance Ratio (CAR)

Former studies did not consider the anti- and vague directional states, however they can be observed frequently when a vehicle tries to stop or reverse direction. The CAR divides between the anti- and vague directional states as shown in Fig. 5. It would be one of the important characteristics of a thruster. At this CAR point, the ambient flow and propeller rotational motion are kept in equilibrium. Hence, to increase the efficiency of the thruster in the reverse thrust mode, an advance ratio value larger than the CAR is preferable, as shown in Fig. 6.

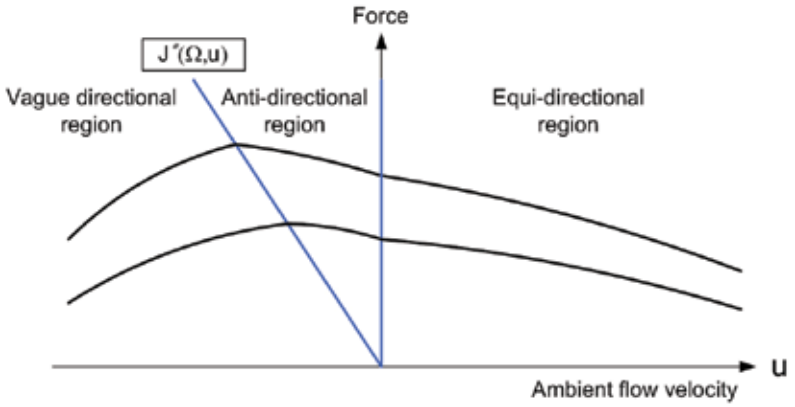


Fig. 6. Thrust force as a function of ambient flow velocity

3.2 Effects of incoming angle on thrust force

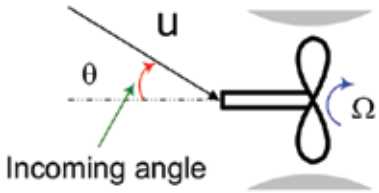


Fig. 7. Incoming angle of ambient flow

In this subsection, the incoming angle effect on thrust force is analyzed. Figure 7 shows the definition of incoming angle. Naturally, if the angle between ambient flow and thrust force is non-parallel, the thrust force varies with the incoming angle. Basically, by multiplying ambient flow velocity by the cosine of the incoming angle, the thrust force can be derived from Eq. (13). In that case, however, the calculated thrust force will not coincide with experimental results except at 0 and 180 degrees, which shows that the incoming angle and ambient flow velocity have another relationship. Hence, we develop the relationship based on experiments, and Fig. 8 shows the result.

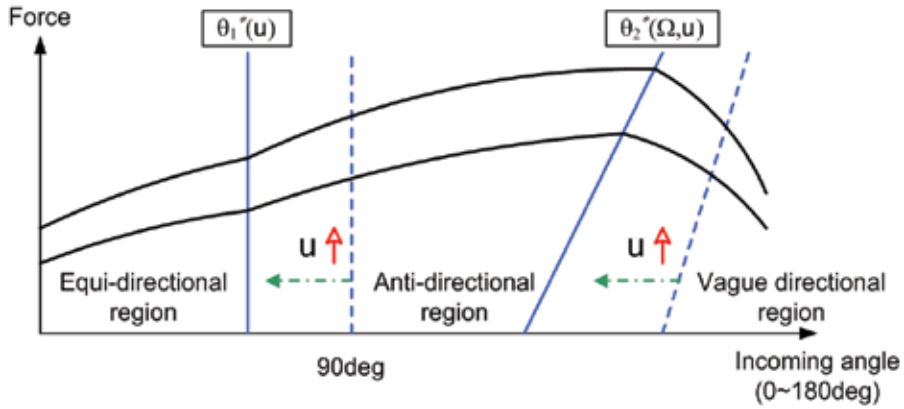


Fig. 8. Thrust force as a function of incoming angle

In Fig. 8, the whole range of angles is also divided into three state regions as denoted in the previous subsection. And, we define the borders of the regions as *Critical Incoming Angles (CIA)* which have the following mathematical relationship.

$$\theta_1^*(u) = \frac{\pi}{2} - a_1 u, \tag{20}$$

$$\theta_2^*(\Omega, u) = a_2 u(\Omega - b_2) + \theta_1^*, \tag{21}$$

where a_1, a_2, b_2, and c_2 are all positive constants. And, $\theta_1^*(u)$ and $\theta_2^*(\Omega, u)$ are the first and second CIA, respectively. Theoretical reasons have not been developed to explain the CIA equations, but empirical results give a physical insight and the above equations can be correlated to experiments. The equi-directional region and anti-directional region are

differentiated with the first CIA. The first CIA only depends on the ambient flow velocity. At the first CIA, the thrust coefficient is the same as the thrust coefficient with no ambient flow velocity. The second CIA separates the anti-directional region and the vague directional region. The second CIA depends not only on ambient flow angle but also on propeller shaft velocity. From Eqs. (20) and (21), the three regions shift to the left as the ambient flow velocity increases.

Now, we derive the incoming angle effect on the thrust force as following:

$$K_T^a = K_T^0 + f_a(J_0, \theta), \tag{22}$$

$$T = K_T^a \rho D^4 \Omega \left|\Omega\right|, \tag{23}$$

where $K_T^0 = K^T(J_0 = 0)$, and

$$f_a = \begin{cases} f_1, & 0 \le \theta \le \theta_1^* \\ f_2, & \theta_1^* \le \theta \le \theta_2^* \\ f_3. & \theta_2^* \le \theta \le \pi \end{cases}; \tag{24}$$

$$\begin{cases} f_1 = (K_T^0 - K_T^+)\left[\sin\left(\dfrac{\theta}{\theta_1^*}\dfrac{\pi}{2}\right) - 1\right] \\ f_2 = K_a J_0 \sin\left(\dfrac{\theta - \theta_1^*}{\pi - \theta_1^*}\dfrac{\pi}{2}\right) \\ f_3 = \left[K_a J_0 \sin\left(\dfrac{\theta_2^* - \theta_1^*}{\pi - \theta_1^*}\dfrac{\pi}{2}\right) - (K_T^- - K_T^0)\right]\cos\left(\dfrac{\theta - \theta_2^*}{\pi - \theta_2^*}\dfrac{\pi}{2}\right) + (K_T^- - K_T^0) \end{cases}$$

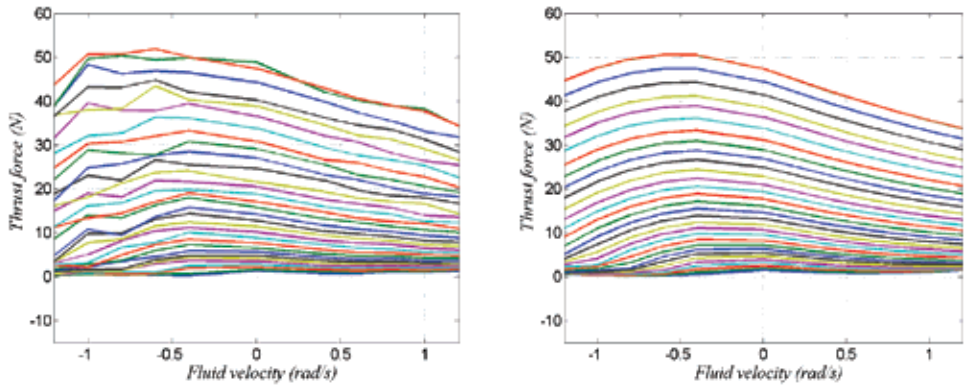
In Eq. (22), K_a is a constant which has to be acquired by experiments. $K_T^+ = K_T(J_0)$ and $K_T^- = K_T(-J_0)$. Eq. (22) coincides with Eq. (13) at 0 and 180 degree. Hence Eq. (22) contains all of the effects of ambient flow and implies that the total thrust force is composed of thrust force with bollard pull condition, K_T^0, and additional force induced by ambient flow velocity and its incoming angle, $f_a(J_0, \theta)$.

4. Experimental results

To verify the proposed model, firstly, we operated the thruster under various ambient flow velocities: ±1.2m/s, ±1.0m/s, ±0.8m/s, ±0.6m/s, ±0.4m/s, and 0m/s with a zero degree incoming angle. Then, for 0.4m/s, 0.6m/s and 0.8m/s ambient flow velocities, the thruster was tilted at 5 degree increments from 0 to 180 degree to change incoming angle. For simplicity, we only consider cases where $\Omega > 0$.

In Fig. 9, the experimental thrust forces are compared with simulation results of the proposed model with an input voltage range from 1.5V to 4.5V, which the whole range is between 0.0V and 5.0V for the positive direction. And the range from 0.0V to 0.8V is dead-

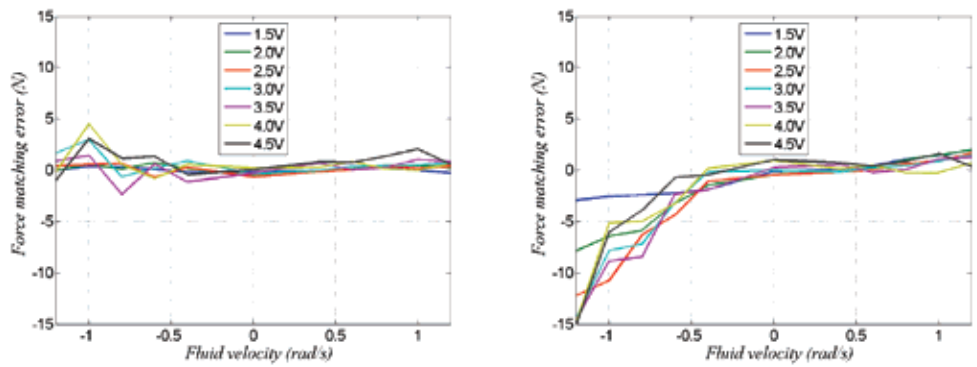
zone. Both results are very similar except at some localized points. The deviation could be caused by the thruster not being located in sufficiently deep water due to the restriction of the experimental environment. Thus, the anti- and vague directional response could have been disturbed by spouting water.



(a) thrust force - experiment (b) thrust force - simulation

Fig. 9. Comparison results of experiment and simulation by the proposed model

To highlight the performance of the proposed model, we compare the results with those of the conventional model described by Eq. (11). The comparison results are shown in Fig. 10. The figure shows that the results of the proposed model are significantly better than the conventional model in the anti- and vague directional regions.

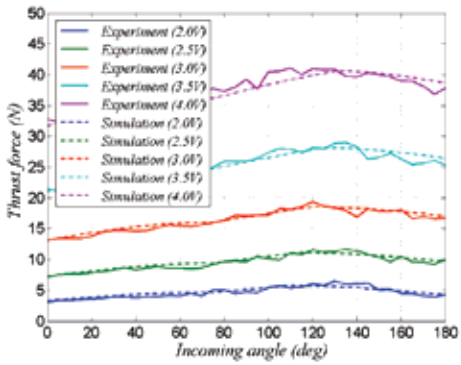


(a) the proposed model (b) the conventional model

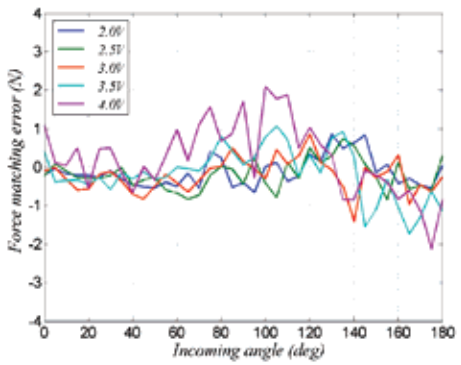
Fig. 10. Thrust force matching error

Figures 11(a), 11(c) and 11(e) show the thrust force comparison between experiment and simulation as a function of incoming angle. The errors of matching, as shown in Figs. 11(b), 11(d) and 11(f), are mostly within ±2N. Note that the maximum force of the thrust is up to 50N.

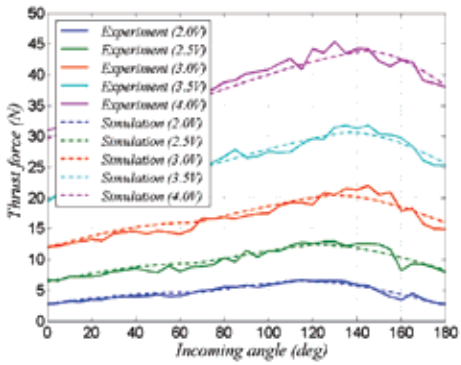
From the matching results with ambient flow velocities and incoming angles, we can say that the initial definition of axial flow is valid, and the proposed model shows good agreement with experimental results under various ambient flow velocities and incoming angles.



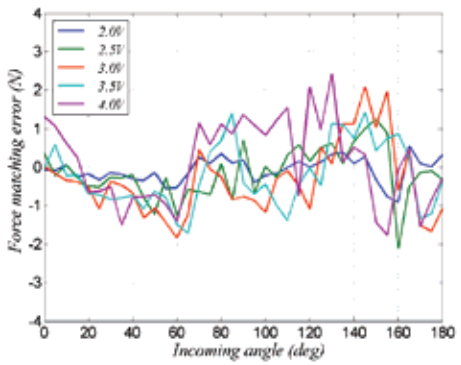
(a) 0.4m/s ambient flow velocity



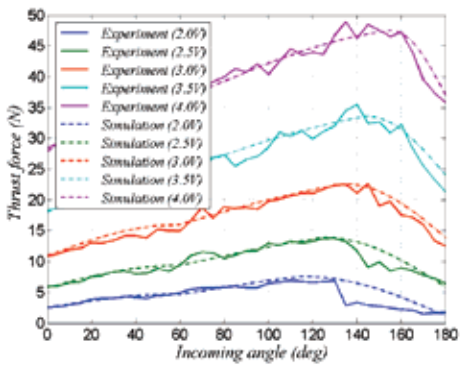
(b) matching error
with 0.4m/s ambient flow velocity



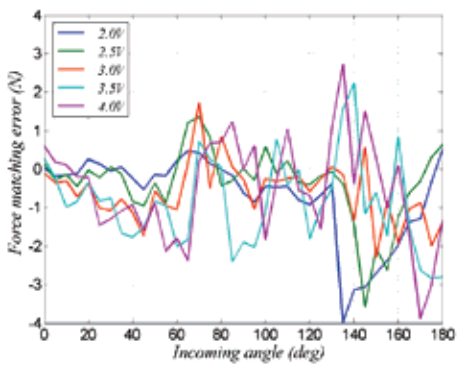
(c) 0.6m/s ambient flow velocity



(d) matching error
with 0.6m/s ambient flow velocity



(e) 0.8m/s ambient flow velocity



(f) matching error
with 0.8m/s ambient flow velocity

Fig. 11. Comparison results of experiment and simulation with incoming angle

5. Thruster controller

5.1 Propeller shaft velocity controller

To obtain the desired thrust force, we need to construct feedback controller with shaft velocity controller. The thruster used in this research only has tachometer for measuring propeller shaft velocity. Hence, firstly, the shaft velocity controller was experimented with open loop and closed loop.

Open Loop (Fig. 12(a))

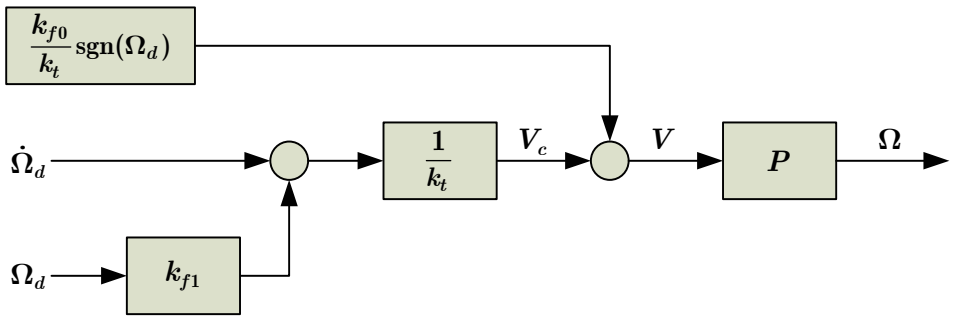
$$V_{in} = \frac{1}{k_t}(\dot{\Omega}_d + k_{f1}\Omega_d) + \frac{k_{f0}}{k_t}\mathrm{sgn}(\Omega_d) \tag{25}$$

Closed Loop (Fig. 12(b))

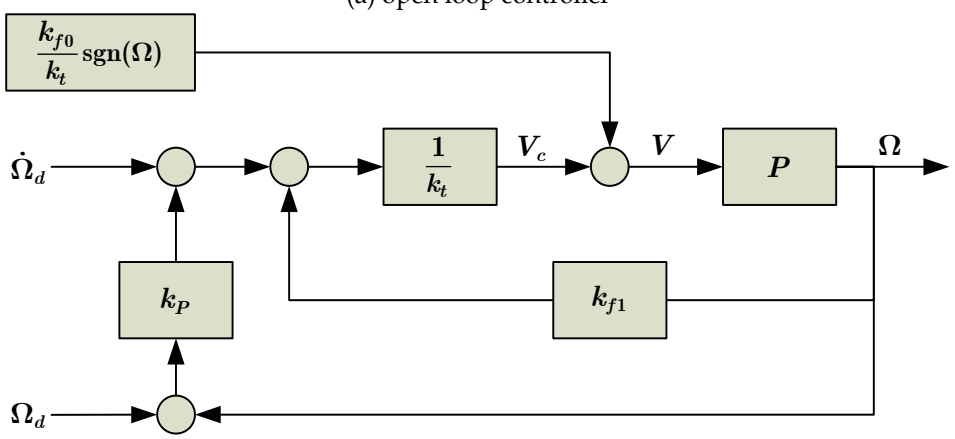
$$V_{in} = \frac{1}{k_t}(\dot{\Omega}_{ref} + k_{f1}\Omega) + \frac{k_{f0}}{k_t}\mathrm{sgn}(\Omega) \tag{26}$$

where

$$\dot{\Omega}_{ref} = \dot{\Omega}_d + k_P(\Omega_d - \Omega) + k_I\int(\Omega_d - \Omega) \tag{27}$$



(a) open loop controller



(b) closed loop controller

Fig. 12. Propeller shaft velocity controller

In Fig. 12, P represents the plant model of thruster and V means the final voltage input to the thruster hardware driver.

5.2 Thrust force controller

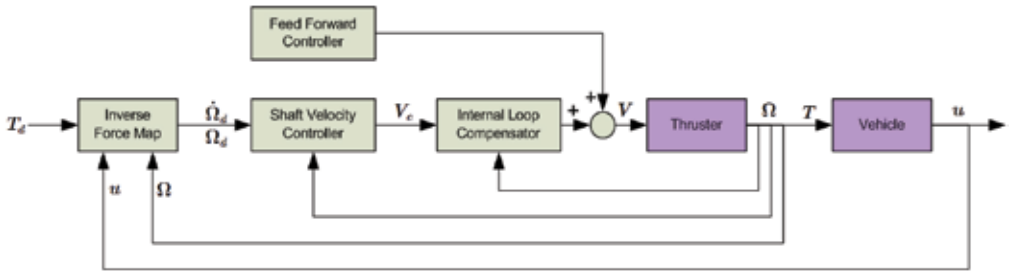


Fig. 13. Diagram of thrust force controller

The overall thrust force controller is composed as Fig. 13. However, the inverse force map only gives the desired shaft velocity according to the desired force. Hence, for the accurate control, the desired shaft acceleration is required. In this chapter, the filtered derivative algorithm is used for the draw of the desired acceleration signal using the desired velocity input.

5.3 Preliminary thruster controller experiments

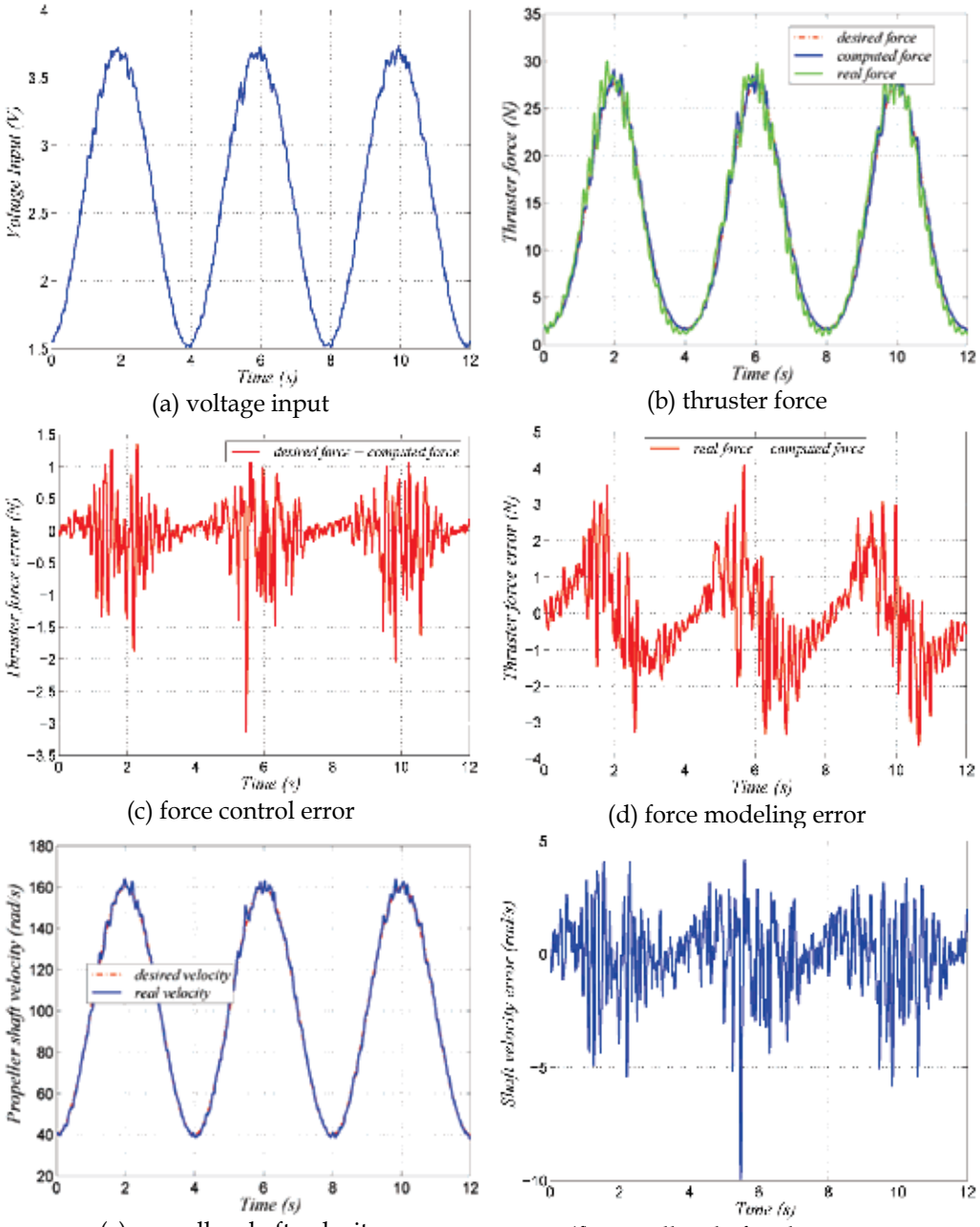
The Bollard-pull condition was tested with open loop and closed loop controller. The closed loop results (Fig. 14) are normally better than open loop results, but the peak error is larger. This comes from the flexible experimental structure. Hence, if the real systems which dose not have structural flexibility, it is expected that the closed loop performance will be better than open loop performance. As shown in the results, the experimental results are good matching with the model. The force control errors are normally less than 5%.

6. Conclusions

In this chapter, a new model of thrust force is proposed. First, the axial flow as a linear combination of the ambient flow and propeller shaft velocity is defined which are both measurable. In contrast to the previous models, the proposed model does not use the axial flow velocity which cannot be measured in real systems, but only uses measurable states, which shows the practical applicability of the proposed model. The quadratic thrust coefficient relation derived using the definition of the axial flow shows good matching with experimental results.

Next, three states, the equi-, anti-, and vague directional states, are defined according to advance ratio and axial flow state. The discontinuities of the thrust coefficient in the non-dimensional plot can be explained by those states. Although they have not been treated previous to this study, the anti- and vague directional states occur frequently when a vehicle stops or reverses direction. The anti- and vague directional states are classified by CAR (Critical Advance Ratio), which can be used to tune the efficiency of the thruster.

Finally, the incoming angle effects to the thrust force, which are dominant in turning motions or for omni-directional underwater vehicles, are analyzed and CIA (Critical Incoming Angle) was used to define equi-, anti-, and vague directional regions.



(a) voltage input (b) thruster force

(c) force control error (d) force modeling error

(e) propeller shaft velocity (f) propeller shaft velocity error

Fig. 14. Experimental thruster control performance of closed loop

The matching results between simulation with experimental results show excellent correlation with only ±2N error in the entire space of thrust force under various ambient flow velocities and incoming angles. Note that the maximum force of the thrust is up to 50N. The results are also compared with conventional thrust models, and the matching

performance with the proposed model is several times better than those of conventional linear ones.

Also in this article, the thrust force control performance of the proposed thruster model was examined. From the results in section 5, the best performance can be obtained by the open loop control with accurate model, because the thrust force cannot be measured directly. This means the force map from the propeller shaft velocity to thrust force plays important roll in control performance. The control performance with the model is acceptable for overall situation, which denoted normally less than ±3N control error.

7. Future works

The thruster modeling and control algorithm need to be enhanced in following aspects.

- Near dead-zone region modeling with complementary experiments
- Dead-zone controller

To precise dynamic positioning control of unmanned underwater vehicles, the dead-zone model and control algorithms should be developed.

8. References

Bachmayer, R. & Whitcomb, L. L. (2003). Adaptive parameter identification of an accurate nonlinear dynamical model for marine thrusters, *J. of Dynamic Sys., Meas., and Control,* Vol.125, No.3, 491–494.

Bachmayer, R.; Whitcomb, L. L. & Grosenbaugh, M. A. (2000). An accurate four-quadrant nonlinear dynamical model for marine thrusters: Theory and experimental validation, *IEEE J. Oceanic Eng.,* Vol.25, No.1, 146–159.

Blanke, M.; Lindegaard, K.-P. & Fossen, T. I. (2000). Dynamic model for thrust generation of marine propellers, In: *IFAC Conf. Maneuvering and Control of Marine Craft (MCMC'2000),* pp. 23–25.

Choi, S. K.; Yuh, J. & Takashige, G. Y. (1995). Development of the omnidirectional intelligent navigator. IEEE *Robotics and Automation Magazine,* Vol.2, No.1, 44–53.

Fossen, T. I. & Blanke, M. (2000). Nonlinear output feedback control of underwater vehicle propellers using feedback form estimated axial flow velocity. *IEEE J. Oceanic Eng.,* Vol.25, No.2, 241–255.

Healey, A. J.; Rock, S. M.; Cody, S.; Miles, D. & Brown, J. P. (1995). Toward an improved understanding of thruster dynamics for underwater vehicles. IEEE J. Oceanic Eng., Vol.20, No.4, 354–361.

Manen, J. D. V. & Ossanen, P. V. (1988). *Principles of Naval Architecture, Second Revision, Volume II: Resistance, Propulsion, and Vibration,* Soc. of Naval Architects and Marine Engineers, ISBN, Jersey City, NJ.

Newman, J. N. (1977). *Marine Hydrodynamics,* MIT Press, Cambridge, MA.

Saunders, A. & Nahon, M. (2002). The effect of forward vehicle velocity on through-body AUV tunnel thruster performance. In: *IEEE/MTS OCEANS '02,* pp. 250–259.

Whitcomb, L. L. & Yoerger, D. R. (1999a). Development, comparison, and preliminary experimental validation of nonlinear dynamic thruster models. *IEEE J. Oceanic Eng.,* Vol.24, No.4, 481–494.

Whitcomb, L. L. & Yoerger, D. R. (1999b). Preliminary experiments in model-based thruster control for underwater vehicle positioning. *IEEE J. Oceanic Eng,* Vol. 24, No.4, 495–506.

Yoerger, D. R.; Cooke, J. G. & Slotine, J.-J. E. (1990). The influence of thruster dynamics on underwater vehicle behavior and their incorporation into control system design. *IEEE J. Oceanic Eng,* Vol.15, No.3, 167–178.

14

Adaptive Control for Guidance of Underwater Vehicles

Mario Alberto Jordán and Jorge Luis Bustamante
Argentinian Institute of Oceanography (IADO-CONICET), National University of the South (UNS), Bahía Blanca Argentina

1. Introduction

Underwater vehicles are extensively employed in the offshore industry, subaquatic scientific investigations and rescue operations. They are sophisticated mechanisms with complex nonlinear dynamics and large lumped perturbations. They can remotely be operated or eventually autonomously navigate along specified scheduled trajectories with geometric and kinematic restrictions for obstacle avoidance or time-optimal operations (Fossen, 1994).
In a wide spectrum of applications, underwater vehicles are generally described by nonlinear and time-varying dynamics. For instance, dynamics with variable inertia and buoyancy arriving from sampling missions or hydrodynamics related to large changes of operation velocity or current perturbations in which laminar-to/from-turbulent transitions are involved in the hydrodynamics.
Due to the inherent nonlinear equations of motions, perturbed environments and complex missions, subaquatic vehicles require the guidance by means of complex controllers that usually involve automatic speed controls, dynamic positioning and tracking, and autopilot systems for automatic steering of depth and altitude. It is experimentally corroborated that adaptive techniques may provide superior trajectory tracking performance compared with the fixed model-based controllers (Smallwood & Whitcomb, 2003; 2004).
Many different adaptive and robust adaptive approaches for underwater vehicles have been discussed in the literature in the past 15 years to handle uncertainties related to the dynamics, hydrodynamics and external disturbances, see for instance Fossen & Fjellstad, 1995; Hsu et al., 2000, Antonelli et al., 2004; Wang & Lee, 2003; Do et al., 2004. However, the employment of novel high-performance nonlinear control design methodologies like backstepping (Krstić et al., 1995), passivity-based approaches (Fradkov et al., 1999) or sliding modes (Hsu et al., 2000) do not appear in the literature except as incipient applications, see for instance, Do & Pan, 2003; Li et al., 2004; Jordán & Bustamante, 2006; 2007; Conte & Serrani, 1999.
From previous theoretic results and some experimental corroborations, it seems that novel adaptive techniques can give rise to an improvement of the global performance in path tracking, above all when more precise manoeuvrability with a high celerity in motion is necessary in a changeable and uncertain subaquatic environment.

In this chapter we present a general adaptive approach based on speed-gradient techniques that are modified for complex time-varying dynamics. Moreover, the dominant vehicle dynamics and hydrodynamics together with the often neglected thruster dynamics are considered in a complete time-varying and nonlinear model.
The chapter is organized as follows. First a description of the vehicle dynamics and hydrodynamics under environmental perturbations in 6 degrees of freedom is given. Additionally, thruster dynamics is described as embedded in the dominant dynamics like a fast dynamics (parasitics). Then, the tracking and regulation problems are introduced in a general form as minimization of a energy cost functional involving positioning and kinematic errors. Afterwards, a design of a fixed controller is presented. The same methodology is extended to the adaptive case. Then, an extension of the adaptive system structure by means of a state/disturbance observer is developed. Afterwards, global convergence of positioning and kinematic path errors is proved in form of theorem results. Finally, the analysis of a selected case study of navigation in a complex sampling mission illustrates the achievable high performance of the presented control approach.

2. Vehicle dynamics

We consider a Lagrangian approach for describing the vehicle dynamics in 6 degrees of freedom (Fossen, 1994). Moreover, we shall develop here an extension of the usual model for time-varying vehicle dynamics.

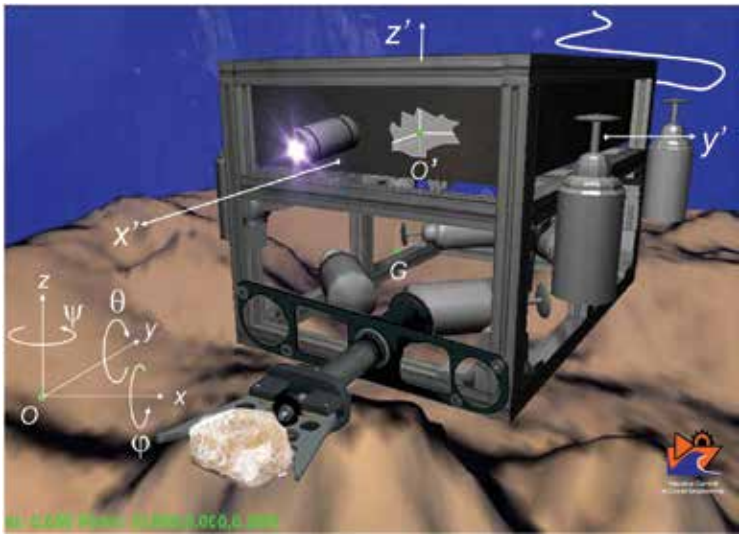


Fig. 1. Case study: full controllable underwater vehicle with 8 thrusters (MUM-TUHH, Hamburg, Germany) with manipulator

2.1 Time-varying dynamics

Consider the Fig. 1. and let us define the generalized position of the vehicle with respect to an earth-fixed frame. This is denoted by the vector $\boldsymbol{\eta} = [x, y, z, \varphi, \theta, \psi]^T$ indicating translations in the x direction (surge), in the y direction (sway), in the z direction (heave), and the so-called Euler angles: φ (roll) about the x axis, θ (pitch) about the y axis, and ψ

(yaw) about the z axis, respectively. Similarly, but referred to a body-fixed frame, the generalized velocity vector $\mathbf{v} = [u, v, w, p, q, r]^T$ indicates linear rates u, v, w, along the main vehicle axis x', y', z', respectively, and angular rates p, q, r, about the axis x', y', z', respectively.

The start point for dynamics description is the kinematic and potential energies of the vehicle, termed T and V, respectively. Consider besides the earth-fixed frame. So, the Lagragian is

$$L = T - V. \tag{1}$$

After applying the Lagragian equation it is valid particularly

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\boldsymbol{\eta}}}\right) - \frac{\partial L}{\partial \boldsymbol{\eta}} + \frac{\partial P_d}{\partial \dot{\boldsymbol{\eta}}} = \boldsymbol{\tau}_\eta, \tag{2}$$

where $\boldsymbol{\tau}_\eta$ is the generalized force applied to the vehicle in some arbitrary point O' (not necessarily the mass center G, see Fig. 1) and P_d is the dissipated energy which is related to a drag force term through

$$\frac{\partial P_d}{\partial \dot{\boldsymbol{\eta}}} = D_\eta\left(\mathbf{v}, \boldsymbol{\eta}\right)\dot{\boldsymbol{\eta}}. \tag{3}$$

with D_η being the generalized drag matrix with respect to the earth-fixed frame. Moreover, the Lagragian is given by

$$L = T_{rb} + T_f - V, \tag{4}$$

where T_{rb} is the kinematic energy of the rigid body, T_f the kinematic energy of the fluid and V fulfills

$$\frac{\partial V}{\partial \boldsymbol{\eta}} = \mathbf{g}_\eta\left(\boldsymbol{\eta}\right), \tag{5}$$

with $\mathbf{g}_\eta$ being the generalized buoyancy force. Additionally, the total kinematic energy in (4) can be expressed as

$$T = T_{rb} + T_f = \frac{1}{2}\dot{\boldsymbol{\eta}}^T M_\eta\left(\boldsymbol{\eta}\right)\dot{\boldsymbol{\eta}}, \tag{6}$$

with M_η being the generalized inertia matrix with respect to the earth-fixed frame. Thus

$$\frac{\partial L}{\partial \dot{\boldsymbol{\eta}}} = M_\eta\left(\boldsymbol{\eta}\right)\dot{\boldsymbol{\eta}} - \frac{\partial V}{\partial \dot{\boldsymbol{\eta}}} = M_\eta\left(\boldsymbol{\eta}\right)\dot{\boldsymbol{\eta}} \tag{7}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\boldsymbol{\eta}}}\right) = M_\eta\left(\boldsymbol{\eta}\right)\ddot{\boldsymbol{\eta}} + \dot{M}_\eta\left(\boldsymbol{\eta}\right)\dot{\boldsymbol{\eta}}. \tag{8}$$

Employing (4), (6) and (5), one achieves

$$\frac{\partial L}{\partial \boldsymbol{\eta}} = \frac{\partial T}{\partial \boldsymbol{\eta}} - \frac{\partial V}{\partial \boldsymbol{\eta}} = \frac{1}{2}\dot{\boldsymbol{\eta}}^T \frac{\partial M(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}\dot{\boldsymbol{\eta}} - \mathbf{g}(\boldsymbol{\eta}). \tag{9}$$

Besides

$$\dot{M}_\eta(\boldsymbol{\eta}) = \dot{\boldsymbol{\eta}}^T \frac{\partial M_\eta(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}. \tag{10}$$

Finally, (2) can be written over again as

$$M_\eta(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} + \frac{1}{2}\dot{M}_\eta(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} + D_\eta(\mathbf{v},\boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \mathbf{g}_\eta(\boldsymbol{\eta}) = \boldsymbol{\tau}_\eta. \tag{11}$$

From this expression, one can recognize the generalized thrust force τ_η and the so-called generalized matrix of the Coriolis and centripetal force given by

$$C_\eta(\mathbf{v},\boldsymbol{\eta}) = \frac{1}{2}\dot{M}_\eta(\boldsymbol{\eta}). \tag{12}$$

On the other side, the most appropriate form of the equation of motion is related to the body-fixed frame. In order for (11) to be expressed in this frame, let us consider the frame coordinate relation

$$\dot{\boldsymbol{\eta}} = J(\boldsymbol{\eta})\mathbf{v}, \tag{13}$$

where J is the rotation matrix (see Fossen, 1994) depending on the Euler angles θ and φ. The matrix J is not singular as long as the pitch angle fulfills $|\theta| < \pi/2$. So, with

$$\ddot{\boldsymbol{\eta}} = \frac{dJ(\boldsymbol{\eta})}{dt}\mathbf{v} + J(\boldsymbol{\eta})\dot{\mathbf{v}} \tag{14}$$

one obtains

$$\mathbf{v} = J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} \tag{15}$$

$$\dot{\mathbf{v}} = J^{-1}(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} - J^{-1}\frac{dJ(\boldsymbol{\eta})}{dt}J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}. \tag{16}$$

With (14) in (16) and comparing this result with (11), one can identify the matrices of the new dynamics description

$$M_\eta(\boldsymbol{\eta}) = J^{-T}(\boldsymbol{\eta})MJ^{-1}(\boldsymbol{\eta}) \tag{17}$$

$$C_\eta(\mathbf{v},\boldsymbol{\eta}) = J^{-T}(\boldsymbol{\eta})\left[C(\mathbf{v}) - MJ^{-1}(\boldsymbol{\eta})\frac{dJ(\boldsymbol{\eta})}{dt}\right]J^{-1}(\boldsymbol{\eta}) \tag{18}$$

$$D_\eta(\mathbf{v},\boldsymbol{\eta}) = J^{-T}(\boldsymbol{\eta})D(\mathbf{v}) \tag{19}$$

$$\mathbf{g}_\eta(\boldsymbol{\eta}) = J^{-T}(\boldsymbol{\eta})\mathbf{g}(\boldsymbol{\eta}) \tag{20}$$

$$\boldsymbol{\tau}_{\eta} = J^{-T}(\boldsymbol{\eta})\,\boldsymbol{\tau}_{t}, \tag{21}$$

with M, C, D, $\mathbf{g}$ and $\boldsymbol{\tau}_t$ being the generalized matrices and vectors accounting for the inertia matrix, the Coriolis and centripetal matrix, the drag matrix, the buoyancy and the thrust force, respectively, all them with respect to the body-fixed frame.
Replacing (17) of M_η in (10) of $\dot{M}_\eta$ and this result in (12) one accomplishes

$$C_{\eta}(\mathbf{v},\boldsymbol{\eta}) = \frac{1}{2}\left(\frac{dJ^{-T}(\boldsymbol{\eta})}{dt}MJ^{-1}(\boldsymbol{\eta}) + J^{-T}(\boldsymbol{\eta})\,\dot{M}J^{-1}(\boldsymbol{\eta}) + J^{-T}(\boldsymbol{\eta})\,M\frac{dJ^{-1}(\boldsymbol{\eta})}{dt}\right). \tag{22}$$

Now from (22) in (18) one obtains the final expression for the generalized matrix of Coriolis and centripetal force with respect to the body-fixed frame

$$C(t,\mathbf{v}) = \frac{1}{2}\left(J^{T}(\boldsymbol{\eta})\,\frac{dJ^{-T}(\boldsymbol{\eta})}{dt}M + \dot{M} + M\frac{dJ^{-1}(\boldsymbol{\eta})}{dt}J(\boldsymbol{\eta})\right) + MJ^{-1}(\boldsymbol{\eta})\,\frac{dJ(\boldsymbol{\eta})}{dt}. \tag{23}$$

It is noticing from (23) that the supposed time variance of the dynamics at the beginning of the section, leads to the appearance of the new term $\frac{1}{2}\dot{M}$ in comparison with the time invariant model in Fossen, 1994.
Finally, the dynamics of the time-varying dynamics of the underwater vehicle is described by two ordinary differential equations (ODEs)

$$\dot{\mathbf{v}} = M^{-1}(t)\left(-\left(C_c(t,\mathbf{v}) + \frac{1}{2}\dot{M}(t)\right)\mathbf{v} - D(t,\mathbf{v})\mathbf{v} - \mathbf{g}(t,\boldsymbol{\eta}) + \boldsymbol{\tau}_t\right) \tag{24}$$

$$\dot{\boldsymbol{\eta}} = J(\boldsymbol{\eta})\,\mathbf{v}. \tag{25}$$

where the time dependence in the systemmatrices is explicitly declared in the notation. Moreover, it is assumed the existence of $M^{-1}(t)$ uniformly in time, and C_c is the part of C in (23) that does not contain the term $\frac{1}{2}\dot{M}$.

2.2 System matrices description

In the following, a detailed description of expressions for the system matrices M, C, D and $\mathbf{g}$ will be given from a physical point of view.
The inertia matrix M can be decomposed into the so-called body inertia matrix M_b and the added mass matrix M_a which accounts for the surrounded fluid mass. So, it is valid

$$M(t) = M_b(t) + M_a(t). \tag{26}$$

The body inertia matrix is

$$M_b(t) = \begin{bmatrix} m(t) & 0 & 0 \\ 0 & m(t) & 0 \\ 0 & 0 & m(t) \\ 0 & -m(t)z_G(t) & m(t)y_G(t) \\ m(t)z_G(t) & 0 & -m(t)x_G(t) \\ -m(t)y_G(t) & m(t)x_G(t) & 0 \end{bmatrix} \tag{27}$$

$$\begin{bmatrix} 0 & m(t)z_G(t) & -m(t)y_G(t) \\ -m(t)z_G(t) & 0 & m(t)x_G(t) \\ m(t)y_G(t) & -m(t)x_G(t) & 0 \\ I_x(t) & -I_{xy}(t) & -I_{xz}(t) \\ -I_{yx}(t) & I_y(t) & -I_{yz}(t) \\ -I_{zx}(t) & -I_{zy}(t) & I_z(t) \end{bmatrix},$$

where m is the vehicle mass, I_{ij} are the inertia moments with respect to the main axis i, j and x_G, y_G and z_G are the coordinates that determine the distance between the mass center G and the coordinate frame center O'. On the other side, the added inertia matrix is

$$M_a = \left(m_{a_{ij}}(t)\right), \text{for } i,\ j = 1, ..., 6, \tag{28}$$

where the elements m_{aij} are functions of time of the added mass in each simple and cross motion with respect to the body main axis. Additionally, the inertia matrix fulfills $M = M^T > 0$. Similarly, the Coriolis and centripetal matrix can be described by taking the body mass and the added mass effects separately

$$C(t, \mathbf{v}) = C_b(t, \mathbf{v}) + C_a(t, \mathbf{v}), \tag{29}$$

where

$$C_b(t, \mathbf{v}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -m(t)\left(y_G(t)q + z_G(t)r\right) & m(t)\left(y_G(t)p + w\right) & m(t)\left(z_G(t)p - v\right) \\ m(t)\left(x_G(t)q - w\right) & -m(t)\left(z_G(t)r + x_G(t)p\right) & m(t)\left(z_G(t)q + u\right) \\ m(t)\left(x_G(t)r + v\right) & m(t)\left(y_G(t)r - u\right) & -m(t)\left(x_G(t)p + y_G(t)q\right) \end{bmatrix}$$

$$\begin{bmatrix} -m(t)\left(x_G(t)q - w\right) & -m(t)\left(x_G(t)r + v\right) & m(t)\left(y_G(t)q + z_G(t)r\right) \\ m(t)\left(z_G(t)r + x_G p\right) & -m(t)\left(y_G(t)r - u\right) & -m(t)\left(y_G(t)p + w\right) \\ -m(t)\left(z_G(t)q + u\right) & m(t)\left(x_G(t)p + y_G(t)q\right) & -m(t)\left(z_G(t)p - v\right) \\ -I_{yz}(t)q - I_{xz}(t)p + I_{zz}(t)r & I_{yz}(t)r + I_{xz}(t)p - I_{yy}(t)q & 0 \\ 0 & -I_{xz}(t)r - I_{xy}(t)q + I_{xx}(t)p & I_{yz}(t)q + I_{xz}(t)p - I_{zz}(t)r \\ I_{xz}(t)r + I_{xy}(t)q - I_{xx}(t)p & 0 & -I_{yz}(t)r - I_{xy}(t)p + I_{yy}(t)q \end{bmatrix} \tag{30}$$

$$C_a(t, \mathbf{v}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -\sum_{i=1}^{6} m_{a_{3i}}(t)v_i & \sum_{i=1}^{6} m_{a_{2i}}(t)v_i \\ \sum_{i=1}^{6} m_{a_{3i}}(t)v_i & 0 & -\sum_{i=1}^{6} m_{a_{1i}}(t)v_i \\ -\sum_{i=1}^{6} m_{a_{2i}}(t)v_i & \sum_{i=1}^{6} m_{a_{1i}}(t)v_i & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -\sum_{i=1}^{6} m_{a_{3i}}(t)v_i & \sum_{i=1}^{6} m_{a_{2i}}(t)v_i \\ \sum_{i=1}^{6} m_{a_{3i}}(t)v_i & 0 & -\sum_{i=1}^{6} m_{a_{1i}}(t)v_i \\ -\sum_{i=1}^{6} m_{a_{2i}}(t)v_i & \sum_{i=1}^{6} m_{a_{1i}}(t)v_i & 0 \\ 0 & -\sum_{i=1}^{6} m_{a_{6i}}(t)v_i & \sum_{i=1}^{6} m_{a_{5i}}(t)v_i \\ \sum_{i=1}^{6} m_{a_{6i}}(t)v_i & 0 & -\sum_{i=1}^{6} m_{a_{4i}}(t)v_i \\ -\sum_{i=1}^{6} m_{a_{5i}}(t)v_i & \sum_{i=1}^{6} m_{a_{4i}}(t)v_i & 0 \end{bmatrix}. \tag{31}$$

Additionally, $C(t, \mathbf{v})$ can be expanded into a linear combination according to

$$C(t,\mathbf{v}) = \sum_{i=1}^{6} C_i(t). \times C_{v_i}(v_i), \tag{32}$$

where the operator ".×" stays for an element-by-element operation between matrices, C_{v_i} are matrices depending on each single rate $v_i \in \{u, v, w, p, q, r\}$ and C_i contains time-varying coefficients of m, I_{ij}, $m_{a_{ij}}$ and the coordinates of G. The matrices C_{v_i} are constructed from (30) and (31) searching for the elements with v_i only, for instance, $C_{v_1}(v_1) = C_u(u) = [0\ 0\ 0\ 0\ u\ u;\ 0\ 0\ 0\ u\ 0\ u;\ 0\ 0\ 0\ u\ u\ 0;\ 0\ u\ u\ 0\ u\ u;\ u\ 0\ u\ u\ 0\ u;\ u\ u\ 0\ u\ u\ 0]$, and similarly for $C_v(v)$ up to $C_r(r)$. In the same way, $C_1 = [0\ 0\ 0\ 0\ -m_{a_{31}}\ m_{a_{21}};\ 0\ 0\ 0\ m_{a_{31}}\ 0\ (m - m_{a_{1i}});\ 0\ 0\ 0\ -m_{a_{21}}\ (-m + m_{a_{1i}})\ 0;\ 0\ -m_{a_{31}}\ m_{a_{21}}\ 0\ -m_{a_{61}}\ m_{a_{51}};\ m_{a_{31}}\ 0\ (m - m_{a_{11}})\ m_{a_{61}}\ 0\ -m_{a_{41}};\ -m_{a_{21}}\ (-m + m_{a_{11}})\ 0\ -m_{a_{51}}\ m_{a_{41}}\ 0]$, and similarly for C_2 up to C_6.

For the drag matrix, a suitable characterization can be done by superposing a constant matrix D_l accounting for laminar fluid effects and a velocity-depending matrix $D_q\ \mathrm{diag}(|\mathbf{v}|)$ accounting by turbulence effects of the hydrodynamics

$$D(t,|\mathbf{v}|) = D_l(t) + D_q(t)\,\mathrm{diag}(|\mathbf{v}|), \tag{33}$$

where the notation $\mathrm{diag}(|\mathbf{v}|)$ means $\mathrm{diag}(|u|,\ldots,|r|)$.

Finally, the total buoyancy vector is described in details by

$$\mathbf{g}(t,\boldsymbol{\eta}) = \begin{bmatrix} (W(t) - W_w(t))\, s(\theta) \\ -(W(t) - W_w(t))\, c(\theta)\, s(\phi) \\ -(W(t) - W_w(t))\, c(\theta)\, c(\phi) \\ -(W(t)y_G(t) - W_w(t)y_B(t))\, c(\theta)\, c(\phi) + (W(t)z_G(t) - W_w(t)z_B(t))\, c(\theta)\, s(\phi) \\ (W(t)x_G(t) - W_w(t)x_B(t))\, c(\theta)\, c(\phi) + (W(t)z_G(t) - W_w(t)z_B(t))\, s(\theta) \\ -(W(t)x_G(t) - W_w(t)x_B(t))\, c(\theta)\, s(\phi) - (W(t)y_G(t) - W_w(t)y_B(t))\, s(\theta) \end{bmatrix}, \tag{34}$$

where $s(.) = \sin(.)$ and $c(.) = \sin(.)$, W is the vehicle weight and W_w its buoyancy and x_B, y_B and z_B represent the coordinates of the metacenter in the body-fixed frame. Finally $\mathbf{g}$ can be appropriately put into a linear combination as

$$\mathbf{g}(t,\boldsymbol{\eta}) = B_1(t)\mathbf{g}_1(\boldsymbol{\eta}) + B_2(t)\mathbf{g}_2(\boldsymbol{\eta}), \tag{35}$$

where

$$\mathbf{g}_1(\boldsymbol{\eta}) = [\sin\theta, \cos\theta\sin\phi, \cos\theta\cos\phi, \cos\theta\cos\phi, \cos\theta\cos\phi, \sin(\theta)]^T \tag{36}$$

$$\mathbf{g}_2(\boldsymbol{\eta}) = [0, 0, 0, \cos\theta\sin\phi, \sin\theta, \cos(\theta)\sin(\phi)] \tag{37}$$

$$\begin{aligned} B_1(t) = \mathrm{diag}\,(&W(t) - W_w(t), W(t) - W_w(t), W(t) - W_w(t), \\ &W(t)y_G(t) - W_w(t)y_B(t), W(t)x_G(t) - W_w(t)x(t)_B, \\ &W(t)y_G(t) - W_w(t)y_B(t)) \end{aligned} \tag{38}$$

$$B_2(t) = \operatorname{diag}(0,0,0, W(t)z_G(t) - W_w(t)z_B(t), \\ W(t)z_G(t) - W_w(t)z_B(t), W(t)x_G(t) - W_w(t)x_B(t)). \tag{39}$$

Clearly, B_1 and B_2 are matrices with time dependence due to changes in the weight, buoyancy, mass center and metacenter, while $\mathbf{g}_1$ and $\mathbf{g}_2$ are state-dependent vectors.

2.3 On the physical variability of the system parameters

The vehicle dynamics is usually subject to changes according to phenomenological and operation-dependent sources. Generally, for vehicles with high manoeuvrability, the hydrodynamics can change in the form of complex transitions from laminar to turbulent flow and vice versa in the wide range of $\mathbf{v}$ and currents. Moreover, in sampling missions over the sea bottom, the vehicle mass and its distribution in the vehicle is modified in a staggered form. Similarly, for fuel-propelled vehicles, there exists a continuous decreasing of the total mass during the navigation and also complex sloshing phenomena take place that modify the inertia properties in time.

For these scenarios, the inertia matrix can be modelled appropriately by

$$M(t) = M_c(t) + \sum_{k=1}^{n} \Delta M_k h(t - t_k), \tag{40}$$

where M_c is the bounded and continuous part of M and ΔM_k are the steps of sudden changes of M at discrete times $t_k \in \mathcal{S}_{t_k}$, with $k = 1, ..., n$, and $\mathcal{S}_{t_k}$ a set for discrete time points and h is the step function. Correspondingly, the derivative of M accomplishes

$$\dot{M}(t) = \dot{M}_c(t) + \sum_{k=1}^{n} \Delta M_k \delta(t - t_k), \tag{41}$$

with δ the delta function. Physically, the sudden changes ΔM_k accounts only for the mass of the vehicle and not for the surrounded fluid, i.e., only M_b but not M_a in (26) is modified.

The variations of M cause also changes in the Coriolis and centripetal matrix C, see (23), in the form of a piecewise continuous evolution in time, but also C is unbounded at time points $t_k \in \mathcal{S}_{t_k}$. Similarly as before, sudden changes of M only affect C_b of the rigid body in (29) but not C_a of the surrounded fluid. Finally, the buoyancy matrices B_1 and B_2 in (35) evolve bounded and piecewice continuous due to changes in (40).

On the other side, the drag matrices D_l and D_{q_i} in (33) are assumed, in a general form, variable in time, also bounded but continuous.

2.4 Existence and uniqueness of solutions

Let us consider again the vehicle dynamics described by the ODEs (24)-(25) and a period of continuity of $M(t)$, i.e., for $t \in [t_k, t_{k+1})$ with $i = 1, ..., n$ and $t_k \in \mathcal{S}_{t_k}$. Additionally let $\boldsymbol{\eta}(t_0) \in \mathcal{S}_\eta$ and $v(t_0) \in \mathcal{S}_v$ be initial conditions.

Since the force components $C_c\mathbf{v}$, $D\mathbf{v}$ and $\mathbf{g}$ are Lipschitz continuous in $\boldsymbol{\eta} \in \mathcal{S}_\eta$ and $\mathbf{v} \in \mathcal{S}_v$, with the matrices $M_c(t)$, $C_{c_i}(t)$, $D_l(t)$, $D_{q_i}(t)$, $B_1(t)$ and $B_2(t)$ being continuous in this period, and $\boldsymbol{\tau}_l(t)$ is generally piecewise continuous in the period considered, then there exists a new, in general, shorter period T where $\boldsymbol{\eta}(t)$ and $\mathbf{v}(t)$ also exist and are unique in $t \in [t_k, t_k+T) \subseteq$

$[t_k, t_{k+1})$ (see Theorem of Existence and Uniqueness, for instance, in O'Reagan, 1997; Vidyasagar, 1993).
It is clear that $\frac{1}{2}\dot{M}\,\mathbf{v}$ of C in (23) will cause a step in the solution for $\mathbf{v}$ at every $t_k \in \mathcal{S}_{t_k}$ equal to

$$\Delta\mathbf{v}(t_k) = \frac{1}{2}M^{-1}(t_k)\Delta M_k\mathbf{v}(t_k). \tag{42}$$

However, the solution for $\boldsymbol{\eta}$ is continuous Lipschitz as it can be deduced from (25).

Thus, assuming the existence and uniqueness of the solutions $\boldsymbol{\eta}(t)$ and $\mathbf{v}(t)$ in every period $[t_k, t_{k+1})$, the complete solution of the system in $[t_0, \infty]$ is obtained by the composition of all pieces with conditions $\mathbf{v}(t_k)$ at the beginning of every piece equal to

$$\mathbf{v}(t_k)=(I + \frac{1}{2}M^{-1}(t_k)\Delta M_k)\lim_{t\to t_k}\mathbf{v}(t), \tag{43}$$

where $\mathbf{v}(t)$ is the solution obtained in $[t_{k-1}, t_k)$. In all this argumentation, the assumption that $\boldsymbol{\eta} \in \mathcal{S}_\eta$ is made for avoiding singularities of J outside $\mathcal{S}_\eta$ is made for $\eta_5 = \theta = \pm\pi/2$.

3. Thruster dynamics

The generalized thrust $\boldsymbol{\tau}_t$ is applied on O' (see Fig. 1). This force is decomposed into the thruster forces f_j actuating along the thruster axial direction j. All together are described in the propulsion vector $\mathbf{f} = [f_1, f_2, ..., f_n]^T$. Equivalently to $\boldsymbol{\tau}_t$ applied on O', f generates the physical impulses along every thruster shaft for the vehicle motion. They are related by

$$\boldsymbol{\tau}_{t_i} = \mathbf{b}_i f_i = \begin{bmatrix} \cos(\psi_i)\cos(\theta_i) \\ \sin(\psi_i)\cos(\theta_i) \\ -\sin(\theta_i) \\ -\sin(\theta_i)\,a_{y_i} - \sin(\psi_i)\cos(\theta_i)\,a_{z_i} \\ \cos(\psi_i)\cos(\theta_i)\,a_{z_i} + \sin(\theta_i)\,a_{x_i} \\ \sin(\psi_i)\cos(\theta_i)\,a_{x_i} - \cos(\psi_i)\cos(\theta_i)\,a_{y_i} \end{bmatrix} f_i, \tag{44}$$

where a_{x_i}, a_{y_i} and a_{z_i} are coordinates of the thruster propeller i referred to O', θ_i is the angle of the axial direction with respect to y' (generally $\theta_i = 0°$ or $\theta_i = 90°$ for horizontal or vertical thrusters, respectively) and ψ_i is the angle of the axial direction with respect to z'. In vector form it is valid

$$\boldsymbol{\tau}_t = \sum_{i=1}^{n}\mathbf{b}_i f_i = B\mathbf{f}, \tag{45}$$

where B is the matrix containing the vectors $\mathbf{b}_i$ as columns, with $i = 1, ..., n$. The thruster dynamics is characterized as (see Healey, 1995; Pinto, 1996; Fossen, 1994)

$$\mathbf{f} = K_1(|\mathbf{n}|.\mathbf{n}) - K_2(|\mathbf{n}|.\mathbf{v}_a) \tag{46}$$

$$\mathbf{n} = \mathbf{n}_1 + \mathbf{n}_2 \tag{47}$$

$$\mathbf{n}_1 = G_1(s)\,\mathbf{f} \tag{48}$$

$$\mathbf{n}_2 = G_2(s)\,\mathbf{u}_a \tag{49}$$

$$\mathbf{u}_a = G_{PID}(s)(\mathbf{n}_r - \mathbf{n}), \tag{50}$$

where $\mathbf{n}$ is the shaft rate vector, $\mathbf{n}_r$ is the reference of the rpm rate for $\mathbf{n}$, $\mathbf{v}_a$ is the vector with the axial velocities of the thrusters, $\mathbf{u}_a$ is the vector of the armature voltages for each DC motor, $\mathbf{n}_1$, $\mathbf{n}_2$ are auxiliary vectors, the operation ($\mathbf{x}.\mathbf{y}$) represents a new vector obtained by an element-by-element product, and similarly $|\mathbf{n}|$ represents a vector with elements $|n_i|$. The factors K_1 and K_2 in (46) are diagonal gain matrices describing the non-linear characteristic (see Fig. 2), G_1 and G_2 represents diagonal matrices with strictly proper Laplace transfer functions, and similarly, G_{PID} is a diagonal matrix with Laplace transfer functions representing an usual tachometric PID control loop of the DC motors.
Finally, the relation between the thruster output $\mathbf{n}$ and its reference $\mathbf{n}_r$ and the propulsion $\mathbf{f}$ is

$$\mathbf{n} = \frac{G_{PID}(s)G_2(s)}{I + G_{PID}(s)G_2(s)}\mathbf{n}_r + \frac{G_1(s)}{I + G_{PID}(s)G_2(s)}\mathbf{f}. \tag{51}$$

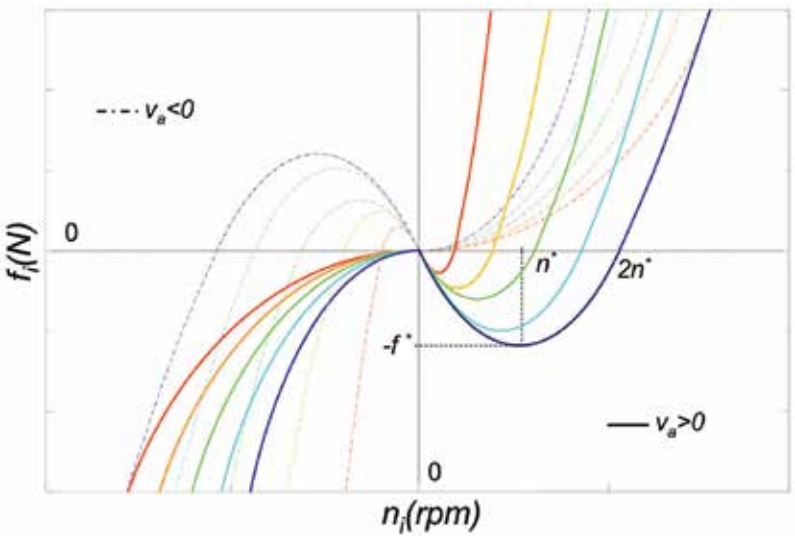


Fig. 2. Static Thruster characteristic

As τ_t is actually the desired control action calculated by some guidance controller, the relation of it with the actuator thrust is also a desired propulsion referred to as

$$\mathbf{f}_{ideal} = B^T \left(BB^T\right)^{-1} \boldsymbol{\tau}_t. \tag{52}$$

In order to reach $\mathbf{f}_{ideal}$, the true f is provided by inputting n_r in (51) to the actuators. This will require the knowledge of $\mathbf{n}$ which is supposed here a non-measurable vector. So, an observer for $\mathbf{n}_r$, $\mathbf{n}$ and $\mathbf{f}$ will be employed later that uses $\mathbf{f}_{ideal}$ and the knowledge of the thruster model to perform the estimations.
Sometimes, the thruster dynamics can be modelled as parasitics in comparison to the dominant vehicle dynamics and consequently be simplified. A typical case of parasitics is

described by vehicles with large inertia for instance. In this case, making $s \to 0$ in (51) one accomplishes $\mathbf{n} = \mathbf{n}_r$ and clearly the thruster is described only by its static characteristic (46) of the Fig. 2. Additionally, using $\mathbf{f}_{ideal}$ and $\mathbf{v}_a$ in (46), $\mathbf{n}_r$ is determined if K_1 and K_2 are known. So, no observer is needed in this case to estimate $\mathbf{f}$ since $\mathbf{f} = \mathbf{f}_{ideal}$.

4. Servo-tracking problem

4.1 Asymptotic path tracking

The path tracking problem for the navigation system is introduced for two reference trajectories $\boldsymbol{\eta}_r(t)$ and $\mathbf{v}_r(t) = J^{-1}(\boldsymbol{\eta}_r)\,\dot{\boldsymbol{\eta}}_r$, which are uniformly continuous. Moreover let $\boldsymbol{\eta}(t)$ and $\mathbf{v}(t)$ be measured vectors that fulfills $\boldsymbol{\eta} \in \mathcal{S}_\eta = \{ x, y, z, \varphi, \psi \in \mathcal{R}^1, |\theta| < \pi/2\}$ and $\mathbf{v} \in \mathcal{S}_v \subset \mathcal{R}^6$. The specifications for the servo-tracking system are

$$\boldsymbol{\eta}(t) - \boldsymbol{\eta}_r(t) \to \mathbf{0}, \text{ for } t \to \infty \tag{53}$$

$$\mathbf{v}(t) - \mathbf{v}_r(t) \to \mathbf{0}, \text{ for } t \to \infty \tag{54}$$

for arbitrary initial conditions $\boldsymbol{\eta}(0) \in \mathcal{S}_\eta$, $\mathbf{v}(0) \in \mathcal{S}_v$. The sets $\mathcal{S}_\eta$ and $\mathcal{S}_v$ describe working regions of stability. Particularly, $\mathcal{S}_\eta$ characterizes the region where $J(\boldsymbol{\eta})$ is non singular (cf. (13) and description below it).

The way to keep spacial and kinematic vehicle trajectories close to their references is achieved by manipulating conveniently the thrust $\mathbf{f}$ by means of a control system. As introduced before, we focus in this work the design of an adaptive control system to achieve this goal.

To this end, let us define first a convenient expression to take account of the positioning and kinematic errors as (Conte & A. Serrani, 1999)

$$\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \boldsymbol{\eta}_r \tag{55}$$

$$\tilde{\mathbf{v}} = \mathbf{v} - J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r + J^{-1}(\boldsymbol{\eta}) K_p \tilde{\boldsymbol{\eta}}, \tag{56}$$

with the gain matrix $K_p = K_p^T \geq 0$. Clearly, from (55)-(56) if $\tilde{\boldsymbol{\eta}}$ is zero with (25), it is valid $\tilde{\mathbf{v}} = \mathbf{v}(t) - \mathbf{v}_r(t)$.

Accordingly, from (24)-(25) with (55)-(56) one obtains the path error system

$$\dot{\tilde{\boldsymbol{\eta}}} = -K_p \tilde{\boldsymbol{\eta}} + J(\boldsymbol{\eta})\tilde{\mathbf{v}} \tag{57}$$

$$\begin{aligned} M\dot{\tilde{\mathbf{v}}} = & -C(\mathbf{v})\,\mathbf{v} - D(|\mathbf{v}|)\mathbf{v} + \mathbf{F}_b(\boldsymbol{\eta}) + \mathbf{F}_c + \mathbf{F}_t - M\frac{d}{dt}\left(J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r\right) + \\ & + M\left(\frac{dJ^{-1}(\boldsymbol{\eta})}{dt}K_p - J^{-1}(\boldsymbol{\eta})K_p^2\right)\tilde{\boldsymbol{\eta}} + MJ^{-1}(\boldsymbol{\eta})K_p J(\boldsymbol{\eta})\tilde{\mathbf{v}}. \end{aligned} \tag{58}$$

Now, we will employ a speed-gradient technique to solve the path tracking problem asymptotically.

4.2 Controller design

The design of speed-gradient controllers starts with the definition of an energy cost functional of the state error vectors $\tilde{\boldsymbol{\eta}}$ and $\tilde{\mathbf{v}}$, which must be constructed as a radially unbounded and nonnegative scalar function (Fradkov et al., 1999). Accordingly, we propose

$$Q(\tilde{\boldsymbol{\eta}}, \tilde{\mathbf{v}}) = \frac{1}{2}\tilde{\boldsymbol{\eta}}^T\tilde{\boldsymbol{\eta}} + \frac{1}{2}\tilde{\mathbf{v}}^T M\tilde{\mathbf{v}}, \tag{59}$$

where Q results piecewise continuous due to the properties of M in time (see Section 2.3).
For an asymptotic stable controlled dynamics it is aimed that for every initial vectors $\boldsymbol{\eta}(0) \in \mathcal{S}_\eta$, $\mathbf{v}(0) \in \mathcal{S}_v$, it is valid from (59) and (53)-(54)

$$Q(\tilde{\boldsymbol{\eta}}(t), \tilde{\mathbf{v}}(t)) \to 0, \text{ for } t \to \infty. \tag{60}$$

According to the speed-gradient method (SG) (Fradkov et al., 1999), the manipulated variable $\boldsymbol{\tau}_t$ has to be build up in that way that $\dot{Q}$ be smooth, bounded and radially decreasing in the error space $\mathcal{S}_{\tilde{\eta}} \times \mathcal{S}_{\tilde{v}}$, and convex in the space of the controller parameters. For the time-varying conditions of the dynamics stated here, $\dot{Q}(t)$ does not fulfills the conditions of continuity because of $M(t)$ in (40).
So, we can analyze $Q(t)$ in the periods of continuity of the trajectory, it is for $t \in [t_0, \infty) \backslash \mathcal{S}_{t_k}$.
Combining (57)-(58), (55)-(56) with (59) and stating the first derivative of $Q(t)$ one obtains

$$\begin{aligned}\dot{Q}(t, t_0, \boldsymbol{\eta}_0, \mathbf{v}_0) = &-\tilde{\boldsymbol{\eta}}^T K_p\tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\eta}}^T J(\boldsymbol{\eta})\tilde{\mathbf{v}} - \tilde{\mathbf{v}}^T C_c(t, \mathbf{v})\,\mathbf{v} - \tilde{\mathbf{v}}^T D_l(t)\mathbf{v} - \\ &-\tilde{\mathbf{v}} D_q(t, |\mathbf{v}|)\,\mathbf{v} - \tilde{\mathbf{v}}^T \mathbf{g}(t, \boldsymbol{\eta}) - \tilde{\mathbf{v}}^T M(t)\,\mathbf{d}(t, \tilde{\boldsymbol{\eta}}, \tilde{\mathbf{v}}) - \\ &-\tilde{\mathbf{v}}^T \frac{\dot{M}_c(t)}{2}\mathbf{v} + \tilde{\mathbf{v}}^T \frac{\dot{M}_c(t)}{2}\tilde{\mathbf{v}} + \tilde{\mathbf{v}}^T \boldsymbol{\tau}_c + \tilde{\mathbf{v}}^T \boldsymbol{\tau}_t\,,\end{aligned} \tag{61}$$

with

$$\mathbf{d}(t, \tilde{\boldsymbol{\eta}}, \tilde{\mathbf{v}}) = \frac{d}{dt}(J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r) - \frac{dJ^{-1}(\boldsymbol{\eta})}{dt}K_p\tilde{\boldsymbol{\eta}} + J^{-1}(\boldsymbol{\eta})K_p^2\tilde{\boldsymbol{\eta}} - J^{-1}(\boldsymbol{\eta})K_p J(\boldsymbol{\eta})\tilde{\mathbf{v}}. \tag{62}$$

With the criterion of eliminating terms with undefined signs in (61) and so achieving the desired properties for $\dot{Q}(t)$, it can be deduced that the following control action is optimal in this sense

$$\begin{aligned}\boldsymbol{\tau}_t(t) = &\sum_{i=1}^{6} U_i \times C_{v_i}(v_i)\mathbf{v} + U_7\mathbf{v} + \sum_{i=1}^{6} U_{i+7}\,|v_i|\,\mathbf{v} + \\ &+U_{14}\mathbf{g}_1 + U_{15}\mathbf{g}_2 + U_{16}\mathbf{d} + U_{17}\tilde{\mathbf{v}} - K_v\tilde{\mathbf{v}} - J^T\tilde{\boldsymbol{\eta}} - \boldsymbol{\tau}_c\,,\end{aligned} \tag{63}$$

where the $\boldsymbol{U}_i$'s are matrices of the controller, C_{v_i} are the system matrices indicated in (32), $\mathbf{g}_1$ and $\mathbf{g}_2$ are the vectors defined in (36) and (37), and finally K_v is a new design matrix involving in the energy of the kinematic errors that accomplishes $K_v = K_v^T \geq 0$.
It is noticing that the cable force $\boldsymbol{\tau}_c$ here is supposed to be measurable and employed directly in the compensation in (63). Moreover, it is important to stress that the $\boldsymbol{U}_i$'s will have the

same structure of null and non-null elements as C_{c1}, ..., C_{c6}, (D_l- $\dot{M}_c$ /2), D_{q1}, ..., D_{q6}, B_1, B_2, M, and $\dot{M}_c$ have, respectively for i = 1, ..., 17. Thus, the number of elements in $\boldsymbol{U}_i$ can be reduced to a minimum considering the system matrix structure. In fact, one simple fixed controller (denoted by U_i^*) could be designed subject to the knowledge of a true dynamics model as

$$U_i^*(t) = C_{c_i}(t), \textit{for } i = 1, ...6 \tag{64}$$

$$U_7^*(t) = D_l(t) + \frac{1}{2}\dot{M}_c(t) \tag{65}$$

$$U_i^*(t) = D_{q_i}(t), \textit{for } i = 8, ...13 \tag{66}$$

$$U_{14}^*(t) = B_1(t) \tag{67}$$

$$U_{15}^*(t) = B_2(t) \tag{68}$$

$$U_{16}^*(t) = M(t) \tag{69}$$

$$U_{17}^*(t) = -\frac{1}{2}\dot{M}_c(t)\ . \tag{70}$$

Besides, the vector function candidate τ_t ($\boldsymbol{U}_i$) in (63) must be such one that Q result convex in the controller parameters in the $\boldsymbol{U}_i$'s. It can be verified from (61)-(62) with (63) that this requirement is in fact satisfied.

4.3 Adaptive controller

Since the system dynamics is unknown, let the controller matrices $\boldsymbol{U}_i$'s be defined adaptively based on speed-gradient laws. Thus, introducing (63) in (61)-(62), it is defined

$$\dot{U}_i = -\Gamma_i \frac{\partial \dot{Q}(U_i)}{\partial U_i},\ \text{for } t \in [t_0, \infty) \backslash \mathcal{S}_{t_k}, \tag{71}$$

with $\Gamma_i = \Gamma_i^T > 0$ constant gain matrices for tuning the adaptation speed in each component $\boldsymbol{U}_i$ of τ_t. These laws allow to obtained $\boldsymbol{U}_i$'s as integral solutions of (71).
Particularly, for the Coriolis and centripetal matrix in its component C_c and its decomposition $C_{c_i}.\times C_{v_i}$ as in (32), it is found for the $\boldsymbol{U}_i$'s with i = 1, ..., 6

$$\dot{U}_i = -\Gamma_i \left(\overset{\frown}{\mathbf{v}}\mathbf{v}^T\right).\times C_{v_i}(v_i). \tag{72}$$

Then, for the linear drag component D_l in (33) the law is applied as

$$\dot{U}_7 = -\Gamma_7 \overset{\frown}{\mathbf{v}}\ \overset{\frown}{\mathbf{v}}^T. \tag{73}$$

Analogously, for the quadratic drag component of D_q in (33) for the $\boldsymbol{U}_i$'s with i = 8, ..., 13, one achieves

$$\dot{U}_i = -\Gamma_i \overset{\frown}{\mathbf{v}} \mathbf{v}^T |v_{i-7}|, \tag{74}$$

where v_i is the i-th element of **v**.
Similarly, for the buoyancy matrices (38) and (39) in (35) it is obtained

$$\dot{U}_{14} = -\Gamma_{14} \overset{\frown}{\mathbf{v}} \mathbf{g}_1^T \tag{75}$$

$$\dot{U}_{15} = -\Gamma_{15} \overset{\frown}{\mathbf{v}} \mathbf{g}_2^T. \tag{76}$$

Next, for the inertia matrix M following law is assigned

$$\dot{U}_{16} = -\Gamma_{16} \overset{\frown}{\mathbf{v}} \mathbf{d}^T, \tag{77}$$

where **d** is the auxiliary vector in (62).
Finally, for the unbounded Coriolis and centripetal component $\frac{1}{2}\dot{M}_c$, one gets

$$\dot{U}_{17} = -\Gamma_{17} \overset{\frown}{\mathbf{v}} \overset{\frown}{\mathbf{v}}^T. \tag{78}$$

The integration of the adaptive laws (72)-(78) with $\boldsymbol{U}_i(t_0) \in \mathcal{S}_U \subset \Re^{6\times 6}$, for i = 1, ..., 17, with $\mathcal{S}_U$ being a compact set, provides a direct calculation of the control action for the path tracking problem without knowledge about the variable system dynamics.

4.4 Modified adaptive laws

With the goal of obtaining a-priori bounded and smooth matrices $\boldsymbol{U}_i(t)$'s, we shall modify (71) with a smooth dynamic projection of the controller matrices (Pomet & Praly, 1992) on every column vector j of $\boldsymbol{U}_i$ referred to as $\mathbf{u}_{ij}$ in the following. Thus, the new $\boldsymbol{U}_i$'s are restrained to

$$\dot{\mathbf{u}}_{ij} = Proy\left(-\left(\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j\right) = \begin{cases} \left(-\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j, \text{ for } \mathbf{u}_{ij} \in \bar{M}_u \text{ or} \\ \left(\nabla_{u_{ij}} \mathcal{P}\right)^T \left(-\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j \leq 0 \\ (I - c(\mathbf{u}_{ij}) \Lambda \frac{\nabla_{u_{ij}} \mathcal{P} \nabla_{u_{ij}} \mathcal{P}^T}{\nabla_{u_{ij}} \mathcal{P} \Lambda \nabla_{u_{ij}} \mathcal{P}^T}) \left(-\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j, \text{ for} \\ \mathbf{u}_{ij} \in M_{u+\varepsilon} \backslash \bar{M}_u \text{or } \left(\nabla_{u_{ij}} \mathcal{P}\right)^T \left(-\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j \leq 0 \end{cases} \tag{79}$$

with

$$c(\mathbf{u}_{ij}) = \min\left\{1, \frac{\mathcal{P}(\mathbf{u}_{ij})}{\varepsilon}\right\}, \tag{80}$$

where $Proy(.)$ denotes the operator of the dynamic projection, additionally $(.)_j$. refers to the column j of the matrix in parenthesis, $\mathcal{P}$ is a convex function in the parametric space of $\mathbf{u}_{ij}$ and is specified in the convex set $M_{u+\varepsilon} = \{\mathbf{u}_{ij} \in \mathcal{R}^6 / \mathcal{P}(\mathbf{u}_{ij}) \leq \varepsilon\}$ composed by the union $M_u = \{\mathbf{u}_{ij} \in \mathcal{R}^6 / \mathcal{P}(\mathbf{u}_{ij}) \leq 0\}$ and the surrounding boundary $M_{u+\varepsilon} \backslash M_u$ with thickness $\varepsilon > 0$, arbitrary small. So $\bar{M}_u$ is the interior of M_u, ∂M_u is the contour of M_u and $\partial M_{u+\varepsilon}$ is the external contour of $M_{u+\varepsilon}$, both supposed smooth. Finally, Λ is a matrix that fulfills $\Lambda = \Lambda^T > 0$.

For future developments of the adaptive controller, it is noticing that $c(\partial M_u)$ and $c(\partial M_{u+\varepsilon})$. Generally, a good choice of $\mathcal{P}(\mathbf{u}_{ij})$ is an hypersphere $\mathcal{P}(\mathbf{u}_{i_j})=\mathbf{u}_{i_j}^T\mathbf{u}_{i_j}-M_0^2-\varepsilon\leq 0$, with $M_0>\varepsilon>0$.

5. Performance and stability of the dominant vehicle dynamics

Let us consider the path tracking problem for the time-varying dynamics of the vehicle and its solution by the adaptive control system described previously. First, let us assume that the thruster dynamics can be neglected, not yet its static characteristic. The actuator parasitics will be considered in the next section. We introduce here the analysis of convergence of tracking error trajectories for the adaptively controlled time-varying dynamics in three steps, namely for system parameters varying: 1) in continuous form, 2) in piecewise-constant form and 3) in piecewise-continuous form. These results are stated by theorems.

5.1 Asymptotic performance

Theorem I *(Asymptotic convergence for time-varying dynamics in continuous form)*
Consider the vehicle system (24)-(25), with bounded, piecewise continuous parameters M, C_{ci}, D_l, D_{qi}, B_1 and B_2, and rates $\dot{M},\dot{C}_{ci},\dot{D}_l,\dot{D}_{qi},\dot{B}_1,\dot{B}_2\in\mathcal{L}_1\cap\mathcal{L}_\infty$ or $\mathcal{L}_2\cap\mathcal{L}_\infty$. Let moreover $\boldsymbol{\tau}_t$ in (63) generated by the direct adaptive controller (72)-(78) with the dynamic projection (79). Assume the thrusters are only described by their nonlinear characteristic in (46). Then, for every initial condition $\boldsymbol{\eta}(t_0)\in\mathcal{S}_\eta$ and $\mathbf{v}(t_0)\in\mathcal{S}_v$, the path tracking problem for given smooth reference trajectories $\boldsymbol{\eta}_r(t)$ and $\mathbf{v}_r(t)$ is achieved asymptotically with null error and the boundness of all variables of the adaptive control loop is ensured if the condition $\boldsymbol{\eta}(t)\in\mathcal{S}_\eta$ is fulfilled for all $t\geq t_0$.

Proof:
Invoking the fact $\boldsymbol{\eta}\in\mathcal{S}_\eta$, eventual singularities of $J(\boldsymbol{\eta})$ are avoided and a Lipschitz condition is guaranteed for the right member of the ODE system (24)-(25) in $\mathcal{S}_\eta\times\mathcal{S}_v$. Then it is assumed the existence and uniqueness of solutions $\boldsymbol{\eta}(t)$ and $\mathbf{v}(t)$ for $t\in[t_0,\infty]$ and arbitrary initial conditions given in the domain of attraction.
Now, let us consider Q in (59) and the following candidate of Lyapunov function

$$V(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i)=Q\left(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}}\right)+Q_1(U_i-U_i^*) \tag{81}$$

where

$$Q_1(U_i-U_i^*)=\frac{1}{2}\sum_{i=1}^{17}\sum_{j=1}^{6}(\mathbf{u}_{i_j}-\mathbf{u}_{ij}^*)^T\Gamma_i^{-1}(\mathbf{u}_{i_j}-\mathbf{u}_{ij}^*) \tag{82}$$

and $\mathbf{u}_{i_j}$ is the column vector j of $\boldsymbol{U}_i$, and analogously, $\mathbf{u}_{i_j}^*$ is the column vector j of the matrix U_i^* in (64)-(70). Then, taking the first derivative of V along the error trajectories $\tilde{\boldsymbol{\eta}}(t)$ and $\tilde{\mathbf{v}}(t)$ in $t\in[t_0,\infty)$ and since $\boldsymbol{\eta}\in\mathcal{S}_\eta$, one accomplishes

$$\begin{aligned}\dot{V}\left(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i\right) &= \dot{Q}\left(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i\right)+\\ &+\sum_{i=1}^{17}\sum_{j=1}^{6}((\mathbf{u}_{i_j}-\mathbf{u}_{i_j}^*(t))^T\Gamma_i^{-1}\dot{\mathbf{u}}_{i_j}-(\mathbf{u}_{i_j}-\mathbf{u}_{i_j}^*(t))^T\Gamma_i^{-1}\dot{\mathbf{u}}_{i_j}^*(t)),\end{aligned} \tag{83}$$

where $\dot{Q}$ *is obtained for the control loop by replacing (63) in (61) for* $t \in [t_0, \infty)$

$$\begin{aligned}\dot{Q}(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i) = &-\tilde{\boldsymbol{\eta}}^T K_p \tilde{\boldsymbol{\eta}} - \tilde{\mathbf{v}}^T K_v \tilde{\mathbf{v}} - \\ &-\tilde{\mathbf{v}}^T \sum_{i=1}^{6} \left(C_{c_i}(t) - U_{i\cdot} \times C_{v_i}(v_i)\right) \mathbf{v} - \tilde{\mathbf{v}}^T \left(D_l(t) + \frac{\dot{M}_c(t)}{2} - U_7\right) \mathbf{v} - \\ &-\tilde{\mathbf{v}}^T \sum_{i=1}^{6} \left(D_{q_i}(t) - U_{i+7}\right) |v_i| \mathbf{v} - \tilde{\mathbf{v}}^T \left(B_1(t) - U_{14}\right) \mathbf{g}_1 + \left(B_2(t) - U_{15}\right) \mathbf{g}_2 - \\ &-\tilde{\mathbf{v}}^T \left(M(t) - U_{16}\right) \mathbf{d} + \tilde{\mathbf{v}}^T \left(\frac{\dot{M}_c(t)}{2} + U_{17}\right) \tilde{\mathbf{v}} \,.\end{aligned} \tag{84}$$

Since $\dot{Q}(\boldsymbol{U}_i)$ *is globally convex in any convex set of the controller parameter space, it is valid*

$$\dot{Q}(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i) - \dot{Q}(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i^*(t)) \leq \sum_{i=1}^{17}\sum_{j=1}^{6} (\mathbf{u}_{i_j}^*(t) - \mathbf{u}_{i_j})^T \frac{\partial \dot{Q}}{\partial \mathbf{u}_{i_j}}, \tag{85}$$

for $t \in [t_0, \infty)$ *and subject to* $\boldsymbol{\eta} \in \mathcal{S}_\eta$. *Replacing (64)-(70) in (84) one obtains* $\dot{Q}(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i^*) = -\tilde{\boldsymbol{\eta}}^T K_p \tilde{\boldsymbol{\eta}} - \tilde{\mathbf{v}}^T K_v \tilde{\mathbf{v}}$.

Now using (85) together with the projected adaptive laws $\dot{\mathbf{u}}_{i_j} = Proy\left(-\left(\Gamma_i \frac{\partial \dot{Q}}{\partial U_i}\right)_j\right)$ *deduced from (72)-(78) with (79), the function (83) can be bounded on the right by*

$$\dot{V}\left(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i\right) \leq -\tilde{\boldsymbol{\eta}}^T K_p \tilde{\boldsymbol{\eta}} - \tilde{\mathbf{v}}^T K_v \tilde{\mathbf{v}} - \sum_{i=1}^{17}\sum_{j=1}^{6} (\mathbf{u}_{i_j} - \mathbf{u}_{i_j}^*(t))^T \Gamma_i^{-1} \dot{\mathbf{u}}_{i_j}^*(t). \tag{86}$$

Using the theorem statement that considers smooth parametric changes and considering (64)-(70), it is valid that $\dot{\mathbf{u}}_{i_j}^*(t)$ *does exist, and that is uniformly continuous and satisfy* $\dot{\mathbf{u}}_{i_j}^* \in \mathcal{L}_1 \cap \mathcal{L}_\infty$ *(and/or* $\dot{\mathbf{u}}_{i_j}^* \in \mathcal{L}_2 \cap \mathcal{L}_\infty$*). Then* $\dot{\mathbf{u}}_{i_j}^* \to 0$ *for* $t \to \infty$ *in* $t \in [t_0, \infty)$ *(see Ioannou & Sun, 1996, Lemma 3.2.5, pp. 76). On the other side, it is valid* $\dot{\mathbf{u}}_{i_j}^* \in \mathcal{L}_\infty$. *Then the projected adaptive law (79) ensures that all* $\mathbf{u}_{i_j}$ *be bounded in* $t \in [t_0, \infty)$. *Finally, integrating (86) with all these suppositions, one achieves*

$$\lim_{t\to\infty} \int_{t_0}^{t} \left(-\tilde{\boldsymbol{\eta}}^T K_p \tilde{\boldsymbol{\eta}} - \tilde{\mathbf{v}} K_v \tilde{\mathbf{v}}\right) d\tau \quad \leq \quad \lim_{t\to\infty} \int_{t_0}^{t} -c(\tau) Q\left(\tau,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}}\right) d\tau \leq \tag{87}$$

$$\leq \quad \lim_{t\to\infty} -c_0 \int_{t_0}^{t} Q\left(\tau,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}}\right) d\tau < \infty, \tag{88}$$

where $c(t)$ *and* c_0 *are positive real values satisfying*

$$c(t) \quad = \quad \frac{\max\limits_{j=1,..,6} \{\lambda_j(K_p), \lambda_j(K_v)\}}{\max\limits_{j=1,..,6} \{\lambda_j(I/2), \lambda_j(M_c(t)/2)\}} \tag{89}$$

$$c_0 = \sup_{t \in \mathcal{S}_t \setminus [0,t_0)} c(t), \tag{90}$$

with $\lambda_j(.)$ representing the eigenvalue j of the matrix indicated in parenthesis. Using the Lemma of Barbalat (Ioannou & Sun, 1996), it follows $\lim_{t\to\infty}(\boldsymbol{\eta}(t)-\boldsymbol{\eta}_r(t)) = \mathbf{0}$. In consequence, with (56) it is also valid

$$\lim_{t\to\infty}(\mathbf{v}-J^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r) = \lim_{t\to\infty}(\mathbf{v}-J^{-1}(\boldsymbol{\eta}_r)\dot{\boldsymbol{\eta}}_r) = \lim_{t\to\infty}(\mathbf{v}-\mathbf{v}_r) = \mathbf{0}. \tag{91}$$

This proves that the path errors go to zero asymptotically as stated in the theorem.
For proving that the $\boldsymbol{U}_i$'s are bounded, one employes (88), (59) and (62) together with the fact that M, $\boldsymbol{\eta}$, $\mathbf{v}$, $\tilde{\boldsymbol{\eta}}$ and J are bounded. Moreover it is valid $\lim_{t\to\infty}\dot{M}, \dot{C}_{c_1}, ..., \dot{C}_{c_6}, \dot{D}_l, \dot{D}_{q_1}, ..., \dot{D}_{q6}, \dot{B}_1, \dot{B}_2 = 0$. So one yields first $\int_{t_0}^{\infty}\left|\tilde{\mathbf{v}}\right|^2 d\tau < \infty$ and $\int_{t_0}^{\infty}\left|\tilde{\mathbf{v}}\right| d\tau < \infty$. Then, from (72)-(77) one obtains

$$|U_i| \le c_i \int_{t_0}^{\infty}\left|\tilde{\mathbf{v}}\right| d\tau < \infty, \text{ for some constant } c_i > 0. \tag{92}$$

Similarly, from (78) it is valid

$$|U_{17}| \le c_{17} \int_{t_0}^{\infty}\left|\tilde{\mathbf{v}}\right|^2 d\tau < \infty, \text{ for some constant } c_{17} > 0, \tag{93}$$

and in this way it is concluded that the matrices $\boldsymbol{U}_i$'s are also bounded. Finally, the boundness of $\boldsymbol{\tau}_c$ in (63) is demonstrated from the boundness of $\tilde{\boldsymbol{\eta}}$ and $\tilde{\mathbf{v}}$, $\mathbf{d}$ in (62), J and the proved boundness of the $\boldsymbol{U}_i$'s. ■

Teorema II *(Asymptotic convergence for piecewise-constant time-varying dynamics)*
Let the statement of Theorem I be valid with the difference that the physical system matrices M, C_{c_v}, B_1 and B_2 are constant for all $t \in [t_0, \infty)/\mathcal{S}_{t_k}$ and that they have finite changes for a finite sequence of isolated time points $t_k \in \mathcal{S}_{t_k}$ with k = 1, ..., n, while the system matrices D_l and D_{q_i} are constant. Then, for every initial condition $\boldsymbol{\eta}(t_0) \in \mathcal{S}_\eta$ and $\mathbf{v}(t_0) \in \mathcal{S}_v$, the path tracking problem for given smooth reference trajectories $\boldsymbol{\eta}_r(t)$ and $\mathbf{v}_r(t)$ is achieved asymptotically with null error and additionally the boundness of all variables of the adaptive control loop is ensured if the condition $\boldsymbol{\eta}(t) \in \mathcal{S}_\eta$ is fulfilled for all $t \ge t_0$.

Proof:
As $\dot{Q}(t)$ in (84) is continuous within $[t_{k-1}, t_k)$ with k = 1, ..., n and additionally in the period $[t_n, \infty)$, and that $\boldsymbol{\eta}(t) \in \mathcal{S}_\eta$, it is valid (see (86) with constant matrices in (64)-(69) and U_{17}^*=0, Theorem I)

$$\dot{V}(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i) \le -\tilde{\boldsymbol{\eta}}^T K_p \tilde{\boldsymbol{\eta}} - \tilde{\mathbf{v}}^T K_v \tilde{\mathbf{v}} < \infty, \tag{94}$$

for $t \in [t_{k-k}, t_k)$ and $t \in [t_n, \infty)$. Employing the solution $\mathbf{v}(t)$ in $[t_{n-1}, t_n)$ and according to (43) and (56) one obtains

$$\tilde{\mathbf{v}}(t_n) = (I+\frac{1}{2}M^{-1}(t_n)\Delta M_n)\lim_{t\to t_n}\mathbf{v}(t)-J^{-1}(\boldsymbol{\eta}(t_n))\dot{\boldsymbol{\eta}}_r(t_n)+J^{-1}(\boldsymbol{\eta}(t_n))K_p\tilde{\boldsymbol{\eta}}(t_n) \in \mathcal{S}_v. \tag{95}$$

Since $\tilde{\boldsymbol{\eta}}(t_n)$ *and* $\tilde{\mathbf{v}}(t_n)$ *are in the attraction domain of the error system in the ODEs, it yields*

$$\lim_{t\to\infty}\int_{t_n}^{t}\dot{V}(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i)d\tau \le \lim_{t\to\infty} -c_0\int_{t_n}^{t} Q\left(\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}}\right)d\tau < \infty, \tag{96}$$

and by Lemma of Barbalat $\lim_{t\to\infty} Q = 0$ *and consequently also the errors* $\tilde{\mathbf{v}}$ *and* $\tilde{\boldsymbol{\eta}}$ *tend asymptotically to zero.*

To demonstrate the boundness of $\boldsymbol{U}_i$*, one notes first that the solutions* $\tilde{\boldsymbol{\eta}}$ *are uniformly continuous while the solutions* $\tilde{\mathbf{v}}$ *are only bounded in* $t < t_n$*. Thus, it is valid* $\int_{t_0}^{t_n}\left|\tilde{\mathbf{v}}\right| dt < \infty$ *and* $\int_{t_0}^{t_n}\left|\tilde{\mathbf{v}}\right|^2 dt < \infty$*, resulting* $|\ \boldsymbol{U}_i\ | < \infty$ *in this period. Then, from* $t_n < \infty$ *in advance, both solutions are bounded and continuous, and so the boundness of the matrices* $\boldsymbol{U}_i$*'s in* $[t_n, \infty]$ *is guaranteed. Finally,* τ_c *in (63) is also bounded as it can be deduced from the boundness of* $\tilde{\boldsymbol{\eta}}$ *and* $\tilde{\mathbf{v}}$*, of* $\mathbf{d}$ *in (62) and J together with the proved boundness of the* $\boldsymbol{U}_i$*'s.* ■

Corollary I *(Asymptotic convergence for staggered-continuous time-varying dynamics)*
Let the statement of Theorem I be valid with the difference that the physical system matrices M, C_{c_i}*, B1 and B2 are piecewise continuous and that* D_l *and* D_{q_i} *are continuous for all* $t \in [t, \infty]$*. The sudden bounded changes occur at* $t_k \in \mathcal{S}_{t_k}$ *with k = 1, ..., n. Additionally, all changes are* $\mathcal{L}_1 \cap \mathcal{L}_\infty$ *or* $\mathcal{L}_2 \cap \mathcal{L}_\infty$*. Then the asymptotic convergence of the tracking errors and the boundness of the adaptive control variables are ensured if the condition* $\boldsymbol{\eta}(t) \in \mathcal{S}_\eta$ *is satisfied for all* $t \ge t_0$*.*

Proof:
The proof of this corollary rises directly from the combination of the results in Theorem I and II in intervals of continuity $[t_{k-1}, t_k)$ *and then, after the last step of the trajectory* $\tilde{\mathbf{v}}(t)$*, by considering the asymptotic disappearance of the parameter variation.* ■

Now the static characteristic of the actuators in (46) is considered in the convergence analysis (see also Fig. 2). It is remembering from the end part of Section 3 that by neglected thruster dynamics it is valid $\mathbf{n} = \mathbf{n}_r$ and $\mathbf{f} = \mathbf{f}_{ideal}$.

Corollary II *(Asymptotic performance of the adaptive control with static thruster characteristic)*
The adaptive control system employed in the path tracking problem for time-varying dynamics with piecewice continuous system parameters preserves the asymptotic properties and the boundness of all variables in the control loop, when the thrusters are described by (46) and (52).

Proof:
As $\mathbf{v}(t)$ *is piecewise continuous, so are* $\mathbf{v}_a(t)$ *in (46),* $\tau_t(t)$ *and* $\mathbf{f}_{ideal}\ (t)$ *in (52). Thus, considering the static characteristic, it is deduced that by solving (46) for* $\mathbf{n}_r(t)$ *with* $\mathbf{f}_{ideal}\ (t)$ *and* $\mathbf{v}_a(t)$*, there may exist one or two solutions at* $t = t_k \in \mathcal{S}_{t_k}$*, each one producing a propulsion that fulfills* $\mathbf{f}_{ideal} - \mathbf{f} = 0$ *in all* $t > t_0$*. Therefore, adopting some criterion in the case of multiplicity of* $\mathbf{n}_r(t_k)$*, one obtains from the last instant* t_n *of sudden change in advance that* $\tilde{\mathbf{v}}(t)$ *will evolve continuously from* $\tilde{\mathbf{v}}(t_n)$ *to zero asymptotically when* $t\to\infty$*. Moreover, due to the dynamic projection, all variables in the adaptive control loop result bounded in the intervals of continuity and also at* t_k*, independent of the multiple solutions for* $\mathbf{n}_r(t_k)$ *and of the temporal variation rate of the parameters so long as they vanish in time.*
■

5.2 Transient performance

The last results has concerned the asymptotic performance of the adaptive control system. However, nothing could be concluded about the control performance in short terms, it is,

about how significant are the transients and how fast the guidance system can adapt the initial uncertainty as well as the temporary changes of the dynamics. Finally, in the focus of future analysis, there would be the rolls that ad-hoc design parameters in both the control loop (i.e., K_p and K_v) and in the adaptive loop (i.e., the Γ_i's) play in the control performance during adaptation transients. A powerful result is given in the following theorem.

Theorem III *(Transient performance of the adaptive control system)*

Let the statement of Corollary I be considered for a piecewise-continuous time-varying dynamics. Then, after an isolated sudden change of M in $t_1 \in \mathcal{S}_{t_k}$ *and depending on the rate laws of* $\dot{M}_c, \dot{C}_{c_j}, \dot{D}_l, \dot{D}_{q_j}, \dot{B}_1$ ***and*** $\dot{B}_2$, *there exists a time point* $t_t > t_1$ *such that, from this on, the adaptive control system with gains* K_p, K_v *and* Γ_i *that are selected sufficiently large, can track any smooth reference trajectories* $\boldsymbol{\eta}_r$ *and* $\mathbf{v}_r$ *with a path error energy that is lower than a certain arbitrarily small level* $\varepsilon > 0$.

Proof:

At $t_1 \in \mathcal{S}_{t_{k'}}$, *the kinematic reference trajectory fulfills* $\mathbf{v}(t_k)=(I+\frac{M^{-1}}{2}(t_k)\Delta M_1)\lim_{t\to t_k}\mathbf{v}(t)$ *(see (43)). Take this vector value as initial condition for the next piece of trajectory of* $\mathbf{v}$ *and consider (86) for* $t > t_1$. *Then it yields*

$$\dot{V}\left(t,\tilde{\boldsymbol{\eta}},\tilde{\mathbf{v}},U_i\right) \leq -c_K\left(\left|\tilde{\boldsymbol{\eta}}\right|^2+\left|\tilde{\mathbf{v}}\right|^2\right)+c_\Gamma^{-1}\sum_{i=1}^{17}\sum_{j=1}^{6}\left|\mathbf{u}_{i_j}-\mathbf{u}_{i_j}^*(t)\right|\left|\dot{\mathbf{u}}_{i_j}^*(t)\right|. \tag{97}$$

with $c_K = \min_{j=1,..,6}\{\lambda_j(K_p),\lambda_j(K_v)\}$ ***and*** $c_\Gamma = \min_{j=1,..,6;\ i=1,...,17}\{\lambda_j(\Gamma_i)\}$. *As the vectors* $\dot{\mathbf{u}}_{i_j}^*(t) \in \mathcal{L}_1 \cap \mathcal{L}_\infty$ or $\mathcal{L}_2 \cap \mathcal{L}_\infty$, *then these start to decrease after expiring some period referred to as* T_{i_j}. *Thus, for a given* $\varepsilon > 0$ *arbitrarily small, there exists some instant* $t_t > \max_{i,j}\left(T_{i_j}\right) + t_1$, *certain minimum values of* c_K *and* c_Γ *from which on* $\dot{V}(t, \tilde{\boldsymbol{\eta}}, \tilde{\mathbf{v}}, U_i) < 0$ *and the previous inequality satisfies*

$$\varepsilon \geq \left|\tilde{\boldsymbol{\eta}}\right|^2+\left|\tilde{\mathbf{v}}\right|^2 > \frac{1}{c_K c_\Gamma}\sum_{i=1}^{17}\sum_{j=1}^{6}\left|\mathbf{u}_{i_j}-\mathbf{u}_{i_j}^*(t)\right|\left|\dot{\mathbf{u}}_{i_j}^*(t)\right|, \tag{98}$$

for $t \geq t_t$. *Clearly, this result is maintained for all* $t \geq t_t$ *if no new sudden change of M occurs any more.* ■

Corollary III *(Transient performance of the adaptive control with static thruster characteristic)*

The result of Theorem III is preserve if the dynamics model and its adaptive control involve actuators with a static characteristic according to (46) and (52).

Proof:

Since $\mathbf{f} - \mathbf{f}_{ideal}$ *is identically zero for all* $t \geq t_0$, *the true propulsion of the vehicle can exactly be generated according to* $\boldsymbol{\tau}_t(t)$ *by the adaptive control system. So the same conditions of Theorem III are satisfy and the same results are valid for the energy of the path error.* ■

It is seen that the adaptive control system stresses the path tracking property by proper setup of the matrices Γ_i's, not only in the selftuning modus but also in the adaptation phase for time-varying dynamics. This can occur independently of the set of K_p and K_v, whose function is more related to the asymptotic control performance, it is when $|\dot{\mathbf{u}}_{i_j}^*(t)| = 0$.

Moreover, it is noticing that in absence of time-varying parameters the dynamic projection on the adaptive laws (79) does not alter the properties of the adaptive control system since

the terms $(\mathbf{u}_{i_j} - \mathbf{u}^*_{i_j}(t))^T \Gamma_i^{-1} \dot{\mathbf{u}}^*_{i_j}(t)$ in (86) are null. The reason for the particular employment of a projection with a smoothness property on the boundary is just the fact that by time-invariant dynamics the control action will result always smooth.

6. State/disturbance observer

The last part of this work concerns the inclusion of the thruster dynamics together with its static characteristic according to (46)-(51) to complete the vehicle dynamics.

By the computation of $\boldsymbol{\tau}_t$ (t) with a suitable selection of the design matrices K_p, K_v and Γ_i's, it is expected that the controlled vehicle response acquires a high performance in transient and steady states. However, as supposed previously, the thruster dynamics (51) has to be considered as long as this does not look as parasitics in comparison with the achievable closed-loop dynamics. There exist approaches to deal with the inclusion of the thruster dynamics in a servo-tracking control problem that takes $\mathbf{f}_{ideal}$ (or the related $\mathbf{n}$) as reference to be followed by $\mathbf{f}$ under certain restrictions or linearizations of the whole dynamics. A comparative analysis of common approaches is treated in Whitcomb et al., 1999, see also Da Cunha, et al., 1995. Though the existing solutions have experimentally proved to give some acceptable accuracy and robustness, they do not take full advantage of the thruster dynamics and its model structure to reach high performance.

In this work a different solution is aimed that employs the inverse dynamics of the thrusters. In this case, the calculated thrust $\mathbf{f}_{ideal}$ in (52) will be used to be input a state/disturbance observer embedded in the adaptive control system to finally estimate the reference $\mathbf{n}_r$ to the shaft rate vector of the actuators that asymptotically accomplishes the previous goal of null tracking errors stated in (53)-(54), see Fig. 3 for the proposed control approach.

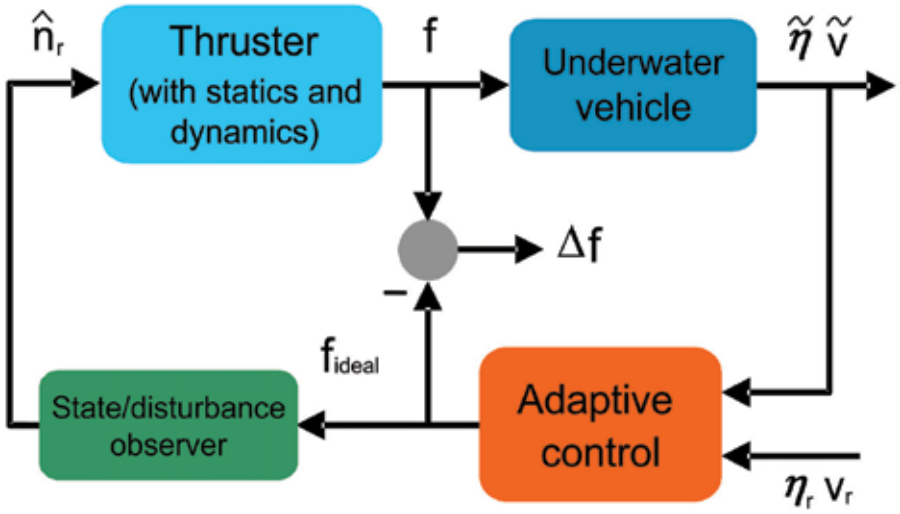


Fig. 3. Extended adaptive control system for the vehicle with thruster dynamics

As start point for an observer design, consider first one element of $G_2G_{PID}(s)$ in (49)- (50) corresponding to one thruster, and a state space description for this dynamics

$$\dot{\mathbf{x}} = A\mathbf{x} + \mathbf{b}(n_r - n) \tag{99}$$

$$n_2 \;=\; n - n_1 = \mathbf{c}^T\mathbf{x} \tag{100}$$

$$\bar{n}_2 \;=\; g_3(s)n - n_1 = -\left(1 - g_3(s)\right)n + \mathbf{c}^T\mathbf{x}, \tag{101}$$

with (A, $\mathbf{b}$, $\mathbf{c}$) a minimal set of a minimal description, x the state of this component and $g_3(s)$ a low pass filter to smooth sudden changes of n. Then, let

$$\dot{\hat{\mathbf{x}}} = A\hat{\mathbf{x}} + \mathbf{b}\,\hat{e}_c + \mathbf{k}_{n_2}(\bar{n}_2 - \hat{n}_2), \tag{102}$$

a differential equation for a state estimation x, with $\mathbf{k}_{n_2}$ a gain vector for the shaft rate error $\bar{n}_2 - \hat{n}_2$ and

$$\hat{n}_2 = \mathbf{c}^T\hat{\mathbf{x}} \tag{103}$$

$$\hat{e}_c = k_n\bar{n}_2 + k_{\dot{n}}\dot{\bar{n}}_2 + \mathbf{k}_{\hat{x}}^T\hat{\mathbf{x}}, \tag{104}$$

with $\hat{n}_2$ and $\hat{e}_c$ estimations of n_2 and of the thruster control error (n_r - n), respectively, and k_n, $k_{\dot{n}}$ and $\mathbf{k}_{\hat{x}}$ suitable gains for the components of $\hat{e}_c$. The function $\bar{n}_2$ and its first derivative $\dot{\bar{n}}_2$ can be deduced and calculated analytically from $\bar{n}_2 = g_3 n - n_1$ with $n_1 = g_1(s)\, f_{ideal}$ and (100). Also with (46) and Fig. 2 with $n^* = \frac{k_2}{2k_1}v_a$, it is valid

$$n = \operatorname{sign}(n - \frac{k_2}{2k_1}v_a)\sqrt{\frac{\operatorname{sign}(n) f_{ideal}}{k_1} + \left(\frac{k_2}{2k_1}v_a\right)^2} + \frac{k_2}{2k_1}v_a. \tag{105}$$

On the other side, with (99), (102), (103) and (101), the state error vector $\breve{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ satisfies

$$\dot{\breve{\mathbf{x}}} = (A - \mathbf{k}_{n_2}\mathbf{c}^T)\breve{\mathbf{x}} + \mathbf{b}\,\tilde{e}_c + \mathbf{k}_{n_2}(1 - g_3(s))n, \tag{106}$$

with $\tilde{e}_c = (n_r - n) - \hat{e}_c$. Using (99)-(100) one gets

$$\dot{n}_2 = \mathbf{c}^T A\mathbf{x} + \mathbf{c}^T\mathbf{b}\,(n_r - n), \tag{107}$$

which combined with (104) and (101) it yields

$$\begin{aligned} \tilde{e}_c \;=\; & \left(1 - k_{\dot{n}}\mathbf{c}^T\mathbf{b}\right)(n_r - n) - \left(k_n\mathbf{c}^T + k_{\dot{n}}\mathbf{c}^T A\right)\mathbf{x} - \\ & - \mathbf{k}_{\hat{x}}^T\hat{\mathbf{x}} + \left(1 - g_3(s)\right)\left(k_n n + k_{\dot{n}}\dot{n}\right). \end{aligned} \tag{108}$$

It is noticed that there exist particular values of the gains $\mathbf{k}_{\hat{x}}$, k_n and $k_{\dot{n}}$ in (108) that fulfills

$$\begin{aligned} & \left(1 - k_{\dot{n}}\mathbf{c}^T\mathbf{b}\right) = 0 \\ & \mathbf{k}_{\hat{x}}^T = -\left(k_n\mathbf{c}^T + k_{\dot{n}}\mathbf{c}^T A\right). \end{aligned} \tag{109}$$

For the state space description in the observer canonical form one has $\mathbf{c}^T = [\,1, 0, \ldots, 0]$ and $\mathbf{b}^T = [b_{m-1}, \ldots, b_0]$. Thus, with the thruster dynamics having a relative degree equal to one (i.e., $b_{m-1} \neq 0$), which is, on the other side, physically true, the observer conditions (109) turns into

$$k_{\dot{n}} = \frac{1}{b_{m-1}} \tag{110}$$

$$\mathbf{k}_{\hat{x}}^T = -\left[\frac{-a_{m-1}}{b_{m-1}} + k_n, \frac{1}{b_{m-1}}, 0, ..., 0\right], \tag{111}$$

with m the system order $G_2G_{PID}(s)$. With these values, (108) can be rewritten as

$$\tilde{e}_c = -\left(k_n \mathbf{c}^T + k_{\dot{n}} \mathbf{c}^T A\right)\tilde{\mathbf{x}} + \left(1 - g_3(s)\right)\left(k_n n + k_{\dot{n}} \dot{n}\right). \tag{112}$$

Additionally, combining (106) with (112) and the choice

$$\mathbf{k}_{n_2} = -k_n \mathbf{b}, \tag{113}$$

it yields

$$\dot{\tilde{\mathbf{x}}} = (I - k_{\dot{n}} \mathbf{b}\mathbf{c}^T) A \tilde{\mathbf{x}} + k_{\dot{n}} \mathbf{b}(1 - g_3(s))\dot{n}, \tag{114}$$

where the matrix $(I - k_{\dot{n}} \mathbf{b}\mathbf{c}^T)A \leq 0$ with only one eigenvalue zero, while (1 - $g_3(s)$) is interpreted as a high-pass filter for the errors $\tilde{\mathbf{x}}$ and $\tilde{e}_c$ that are produced by fast changes of n in order to reach an effective tracking of f_{ideal}.

In order for (114) to give exponentially stable homogeneous solutions $\tilde{\mathbf{x}}(t)$, the first element of the initial condition vector $\hat{\mathbf{x}}(0)$ must be set to null. Moreover, it is noticing that only high-frequency components of $\dot{n}$ can excite the state error dynamics and that these avoid vanishing errors. So, the price to be be paid for including the thruster dynamics in the control approach is the appearance of the vector error $\Delta\mathbf{f}$ which is bounded and its magnitude depends just on the energy of the filtered $\dot{\mathbf{n}}$ in the band of high frequencies. According to (46), the influence of $\dot{\mathbf{n}}$ on $\Delta\mathbf{f}$ is attenuated by small values of the axial velocity of the actuators $\mathbf{v}_a$. In this way, the benefits in the control performance for including the thruster dynamics are significant larger than those of not to accomplish this, i.e., a vehicle model with dominant dynamics only.

Finally, the reference vector $\mathbf{n}_r$ for the inputs of all the thrusters is calculated by means of (104) and (105) in vector form as

$$\hat{\mathbf{n}}_r(t) = \hat{\mathbf{e}}_c(t) + G_3(s)\mathbf{n}(t). \tag{115}$$

The estimation of $\mathbf{n}_r$ closes the observer approach embedded in the extended structure of the adaptive control system described in Fig. 3.

7. Case study

To illustrate the performance of the adaptive guidance system presented in this Chapter, a case study is selected composed on one side of a real remotely teleoperated vehicle described in (Pinto, 1996, see also Fig. 1) and, on the other side, of a sampling mission application over the sea bottom with launch and return point from a mother ship. These results are obtained by numerical simulations.

7.1 Reference path

The geometric reference path $\boldsymbol{\eta}_r$ for the mission is shown in Fig. 4. The on-board guidance system has to conduct the vehicle uniformly from the launch point down 10(m) and to rotate about the vertical line $3/4\ \pi(rad)$ up to near the floor. Then, it has to advance straight 14(m) and to rotate again $\pi/4$ (rad) to the left before positioning correctly for a sampling operation. At this point, the vehicle performs the maneuver to approach 1(m) slant about $\pi/4$ (rad) to the bottom to take a mass of 0.5 (Kg), and it moves back till the previous position before the sampling maneuver. Afterwards it moves straight at a constant altitude, following the imperfections of the bottom (here supposed as a sine-curve profile). During this path the mass center G is perturbed periodically by the sloshing of the load. At this path end, the vehicle performs a new sampling maneuver taking again a mass of 0.5 (Kg). Finally the vehicle moves 1(m) sidewards to the left, rotates 3.535 (rad) to the left, slants up 0.289 (rad) and returns directly to the initial position of the mission. Moreover, the corners of the path are considered smoothed so that the high derivatives of $\boldsymbol{\eta}_r$ exist.

7.2 Design parameters

Here, the adaptive control system is applied according to the structure of the Fig. 3, i.e., with the vehicle dynamics in (24)-(25), the thruster dynamics in (46)-(50) and (52), the control law in (63), the adaptive laws in (72)-(78), and finally the estimation of the thruster shaft rate given in (115). The saturation values for the actuator thrust was set in $\pm 30N$.

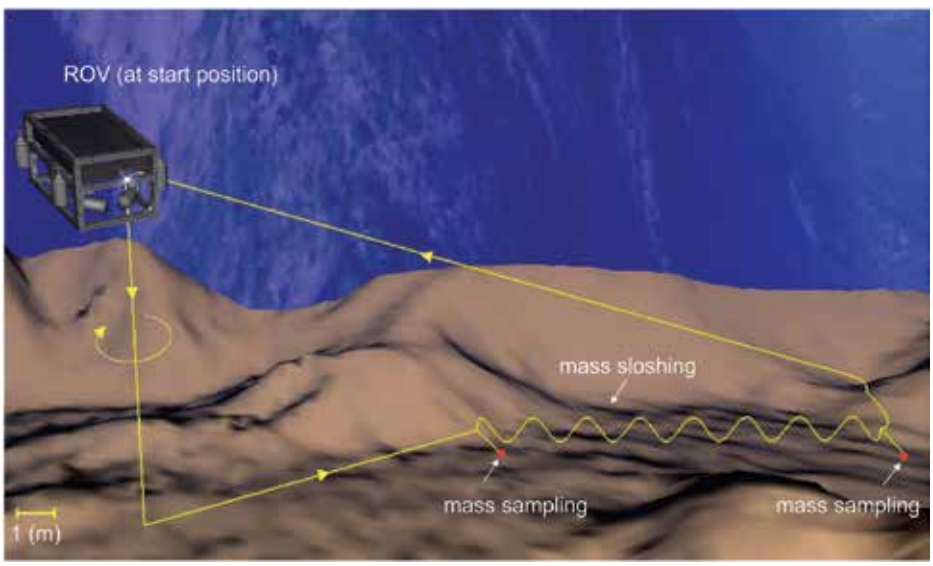


Fig. 4. Case study: sampling mission for an adaptively guided underwater vehicle

Moreover, the controller design gains are setup at large values according to theorem III in order to achieve a good all-round transient performance in the whole mission. These are

$$\begin{aligned} K_p &= \mathrm{diag}(1,1,1,1,1,1) \\ K_v &= \mathrm{diag}(10^3,10^3,10^3,10^3,10^3,10^3) \\ \Gamma_i &= \mathrm{diag}\left(10^2,10^2,10^2,10^2,10^2,10^2\right)\ i=1,...,16 \\ \Gamma_{17} &= \mathrm{diag}\left(1,1,1,1,1,1\right). \end{aligned} \tag{116}$$

Besides, the design parameters for the observer are setup at values

$$\begin{aligned} k_n &= 5 \times 10^{-3} \\ k_{\dot{n}} &= 13.005 \\ \mathbf{k}_{\hat{x}}^T &= [99.538, -13.005, 0] \\ \mathbf{k}_{n_2}^T &= \left[-3.844 \times 10^{-4}, -2.668, -0.207\right] \\ G_3(s) &= \mathrm{diag}\left(\tfrac{5\times 10^6}{(s+1000)(s+5000)}\right). \end{aligned} \tag{117}$$

The main design parameter k_n was chosen roughly in such a way that a low perturbation norm $\| \Delta\mathbf{f} \|_\infty$ in the path tracking and an acceptable rate in the vanishing of the error ($\mathbf{n}_r - \mathbf{n}$) occur. The remainder observer parameters $k_{\dot{n}}$, $\mathbf{k}_{\hat{x}}$ and $\mathbf{k}_{n_2}$ were deduced from the thruster coefficients and k_n according to (110), (111) and (113), respectively. Finally, the battery of filters $\boldsymbol{g}_3(s)$ was selected with a structure like a second-order system.

7.3 Numerical simulations

Now we present simulation results of the evolutions of position and rate states in every mode. The vehicle starts from a position and orientation at rest at $t_0 = 0$ that differs from the earth-fixed coordinate systems in

$$\begin{aligned} &\Delta x(0) = 0.1(m) \quad \Delta y(0) = 0.1(m) \quad \Delta z(0) = 0.1(m) \\ &\Delta\varphi(0) = \tfrac{\pi}{6}(rad) \quad \Delta\theta(0) = \tfrac{\pi}{6}(rad) \quad \Delta\psi(0) = \tfrac{\pi}{4}(rad). \end{aligned} \tag{118}$$

Moreover, the controller matrices $U_i(0)$ are set to null, while no information of the system parameters was available for design aside from the thruster dynamics.

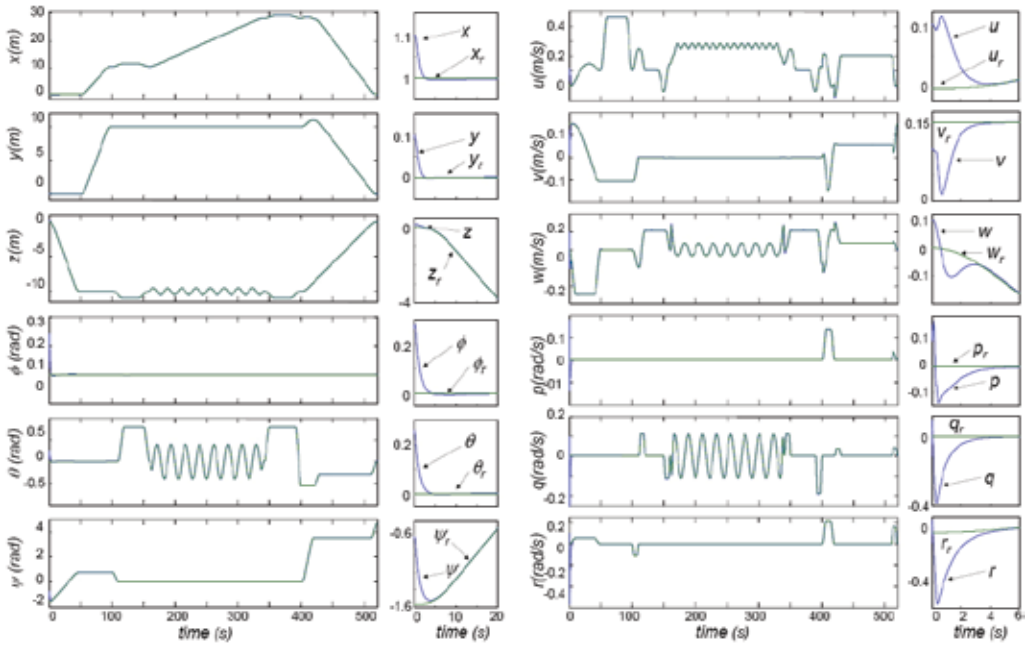


Fig. 5. Path tracking in the position modes ($\boldsymbol{\eta}$ vs. $\boldsymbol{\eta}_r$) (left) and in the kinematic modes ($\mathbf{v}$ vs. $\mathbf{v}_r$) (right)

In Fig. 5 the evolutions of position and kinematics modes are illustrated (left and right, respectively). One sees that no appreciable tracking error occurs during the mission aside from moderate and short transients of about 5(*s*) of duration in the start phase above all in the velocities. During the phase of periodic parameter changes (160 (*s*) up to 340 (*s*)) and at the mass sampling points occurring at 130 (*s*) and 370.5 (*s*), no appreciable disturbance of the tracking errors was noticed. However in the kinematics, insignificant staggered changes were observed at these points and a rapid dissipation of the error energy took place.
The sensibility of time-varying changes in the vehicle dynamics can be perceived above all in the thrust evolution. We reproduce in Fig. 6 the behavior of the eight thrusters of the ROV during the sampling mission; first the four vertical thrusts (2 and 3 in the bow, 1 and 4 in the stern) followed by the four horizontal ones (6 and 7 in the bow, 5 and 8 in the stern) (See Fig. 1). Both the elements of $\mathbf{f}_{ideal}$ and the ones of $\mathbf{f}$ are depicted together (see Fig. 6). It is noticing that almost all the time they are coincident and no saturation occurs in the whole mission time. Aside from the short transients of about 5(*s*) at the start phase, there is, however, very short periods of non coincidence between f and f_{ideal}. For instance, a transient at about 10(s) in the vertical thruster 3 occurs, where a separation in the form of an oscillation of (f - f_{ideal}) less than 4% of the full thrust range is observed (see f_3 and n_3 in Fig. 7, top). This is caused by jumps of the respective shaft rate by crossing discontinuity points around zero of the nonlinear characteristic.

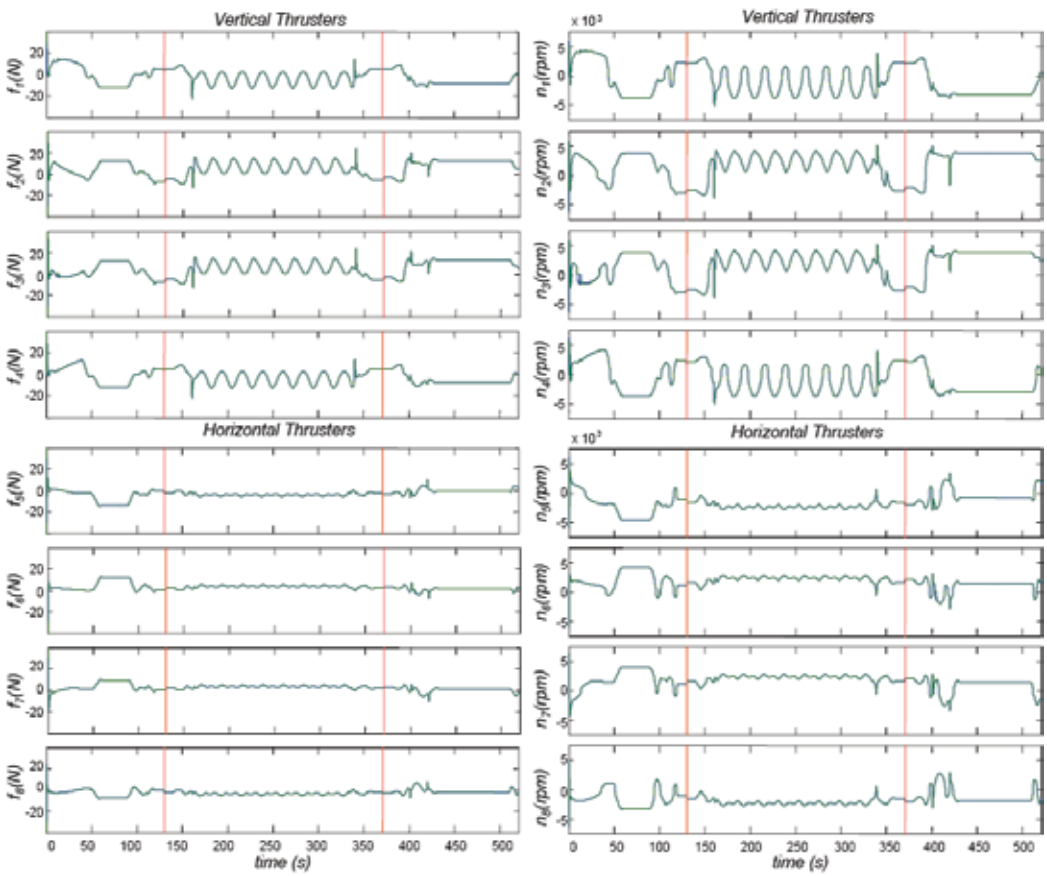


Fig. 6. Evolution of the actuator trusts (**f**) (left) and shaft rates ($\mathbf{n}_t$ vs. $G_3\mathbf{n}_{ideal}$) (right)

Similarly, another short period with the same symptoms and causes takes place in the horizontal thruster 6 at about 404(*s*), also in the form of an oscillation with a separation less that 4% (see f_6 and n_6 in Fig. 7, bottom).

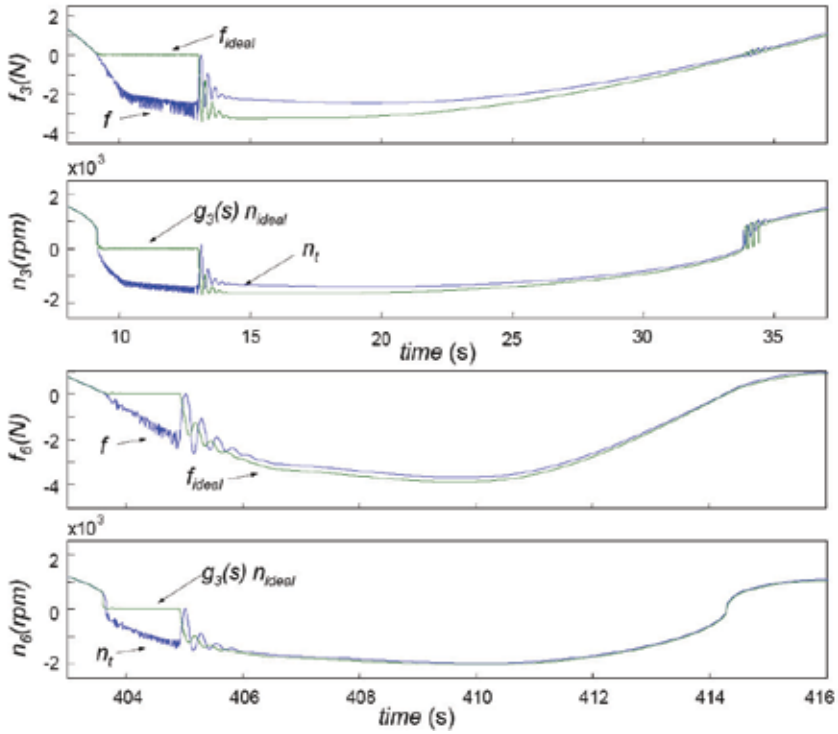


Fig. 7. Evolution of f vs. f_{ideal} and n vs. $g_3 n_{ideal}$ in thruster 3 at about 10(s) (top) and in thruster 6 at about 404(s) (bottom)

The sudden mass changes are absorbed above all by thrusters 2 and 3 (vertical thrusters in the bow) where jumps are also noticed in the evolutions of thrusts. However they have retained an exact coincidence between f and f_{ideal}. Jumps are noticed in all four horizontal thrusters too, with the same amplitude, however to a lesser degree. The coincidence between f and f_{ideal} also persists during periodic parameter changes in all thrusters, see Fig. 6, left.

The performance of the disturbance/state observer can be seen in Fig. 6, right, where the true shaft rate n versus the filtered ideal shaft rate $g_3 n_{ideal}$ are depicted for all thrusters. One notices a good concordance between both evolutions in almost the whole period of the mission. Contrary to the thrust evolutions, the convergence transients of n to $g_3 n_{ideal}$ at the start phase take a very short time less than 1(*s*). However, the evolutions begin with strong excursions and remain in time only a few seconds.

Similarly as in the thrusts f and f_{ideal}, there exist additionally two significant periods with short transients of non coincidence between n and $g_3 n_{ideal}$. These occur at about 10(s) and 404(s) by thrusters 3 and 6, respectively (see Fig. 7, top and bottom). All of them are related to crosses around the zero value under a relatively large value of its axial velocity v_a (cf. Fig. 2). One notices that the evolution of n is more jagged than that of $g_3 n_{ideal}$ due to the discontinuities at the short transients and due to the fact that $g_3 n_{ideal}$ is a smoothed signal.

8. Conclusions

In this chapter a complete approach to design a high-performance adaptive control system for guidance of autonomous underwater vehicles in 6 degrees of freedom was presented. The approach is focused on a general time-varying dynamics with strong nonlinearities in the drag, Coriolis and centripetal forces, buoyancy and actuators. Also, the generally rapid dynamics of the actuators is here in the design not neglected and so a controller with a wide working band of frequencies is aimed.

The design is based on a adaptive speed-gradient algorithm and an state/disturbance observer in order to perform the servo-tracking problem for arbitrary kinematic and positioning references. It is shown that the adaptation capability of the adaptive control system is not only centered in a selftuning phase but also in the adaptation to time-varying dynamics as long as the rate of variation of the system parameter is vanishing in time. Moreover, bounded staggered changes of the system matrices are allowed in the dynamics.

By means of theorem results it was proved that the path-tracking control can achieve always asymptotically vanishing trajectory errors of complex smooth geometric and kinematic paths if the thruster set can be described through its nonlinear static characteristics, i.e., when its dynamics can be assumed parasitic in comparison with the dominant controlled vehicle dynamics and therefore neglected. This embraces the important case for instance of vehicles with large inertia and parsimonious movements. On the other side, when the actuators are completely modelled by statics and dynamics, an observer of the inverse dynamics of the actuators is needed in order to calculate the setpoint inputs to the thrusters. In this case, the asymptotic path tracking is generally lost, though the trajectory errors can be maintain sufficiently small by proper tuning of special ad-hoc high-pass filters. It is also shown, that the transient performance under time-varying dynamics can be setup appropriately and easily with the help of ad-hoc design matrices. In this way the adaptive control system can acquire high-performance guidance features.

A simulated case study based on a model of a real underwater vehicle illustrates the goodness of the presented approach.

9. References

Antonelli, G.; Caccavale, F. & Chiaverini, S. (2004). Adaptive tracking control of underwater vehicle-manipulator systems based on the virtual decomposition approach. *IEEE Trans. on Robotics and Automation*, Vol. 20, No. 3, June 2004, 594-602, ISSN: 1042-296X.

Conte, G. & Serrani, A. (1999). Robust Nonlinear Motion Control for AUVs. *IEEE Rob. and Autom. Mag.*,Vol. 6, No. 2, June 1999, 32-38, ISSN: 1070-9932.

Da Cunha, J.P.V.S; Costa R. R. & Hsu, L. (1995). Design of a High Performance Variable Structure Position Control of ROV's. *IEEE Journal Of Oceanic Engineering*, vol. 20, No. 1, January 1995, 42-55, ISSN: 0364-9059.

Do, K.D. &Pan, J. (2003), Robust and adaptive path following for underactuated autonomous underwater vehicles. *Proceedings of American Control Conference 2003,* pp. 1994- 1999, Denver, USA, 4-6 June 2003.

Do, K.D.; Pan, J. & Jiang, Z.P. (2004). Robust and adaptive path following for underactuated autonomous underwater vehicles. *Ocean Engineering*, Vol. 31, No. 16, November 2004, 1967-1997, ISSN: 0029-8018.

Fossen, T.I. (1994). *Guidance and Control of Ocean Vehicles,* John Wiley&Sons, ISBN: 0- 471-94113-1, Chichester, UK.

Fossen, T.I & Fjellstad, I.E. (1995). Robust adaptive control of underwater vehicles: A comparative study. *Proceedings of the 3rd IFAC Workshop on Control Applications in Marine Systems*, pp. 66-74, Trondheim, Norway, 10-12 May 1995.

Fradkov, A.L.; Miroshnik, I.V. & Nikiforov, V.O. (1999). *Nonlinear and adaptive control of complex systems*, Kluwer Academic Publishers, ISBN 0-7923-5892-9, Dordrecht, The Netherlands.

Healey, A.J.; Rock, S.M.; Cody, S.; Miles, D. & Brown, J.P. (1995). Toward an improved understanding of thruster dynamics for underwater vehicles. *IEEE Journal Of Oceanic Engineering*, vol. 20, No. 4, Oct. 1995, 354–361, ISSN: 0364-9059.

Hsu, L.; Costa, R.R.; Lizarralde, F. & Da Cunha, J.P.V.S. (2000). Dynamic positioning of remotely operated underwater vehicles. *IEEE Robotics & Automation Magazine*, Vol. 7, No. 3, Sept. 2000 pp. 21-31., ISSN: 1070-9932.

Ioannou, P.A. & Sun, J. (1996). *Robust adaptive control*. PTR Prentice-Hall, ISBN: 0-13- 439100-4, Upper Saddle River, New Jersey, USA.

Jordán, M.A. & Bustamante, J.L. (2006). A Speed-Gradient Adaptive Control with State/ Disturbance Observer for Autonomous Subaquatic Vehicles. *Proceedings of IEEE 45th Conference on Decision and Control*, pp. 2008-2013, San Diego USA, 13-15 Dec. 2006.

Jordán, M.A. & Bustamante, J.L. (2007). An adaptive control system for perturbed ROVs in discrete sampling missions with optimal-time characteristics. *Proceedings of IEEE 46th Conference on Decision and Control,* pp. 1300-1305, New Orleans, USA, 12- 14 Dec. 2007.

Jordán, M.A. & Bustamante, J.L. (2007). Oscillation control in teleoperated underwater vehicles subject to cable perturbations. *Proceedings of IEEE 46th Conference on Decision and Control,* pp. 3561-3566, New Orleans, USA, 12-14 Dec. 2007.

Kreuzer, E. & Pinto, F. (1996). Controlling the Position of a Remotely Operated Underwater Vehicle. *App. Math. & Comp.*, Vol, 78, No. 2, September 1996 , 175-185. ISSN: 0096-3003.

Krstić, M.; Kanellakopoulus, I. & Kokotović, P. (1995). *Nonlinear and adaptive control design*, John Wiley and Sons, Inc., ISBN 0-471-12732-9, New York, USA.

Li, J.-H.; Lee, P.-M. & Jun, B.-H. (2004). An adaptive nonlinear controller for diving motion of an AUV, *Proceedings of Ocean '04 - MTS/IEEE Techno-Ocean '04*, pp. 282- 287, Kobe, Japan, 9-12 Nov. 2004.

O'Reagan, D. (1997) *Existence theory for nonlinear ordinary differential equations. Mathematics and its Application*s. Kluwer Academic Publishers, ISBN 0–7923–4511–8, Dordrecht: The Netherlands.

Pinto F. (1996). Theoretische und Experimentelle Untersuchungen zur Sensorik und Regelung von Unterwasserfahrzeugen, Doctoral Thesis.

Pomet, J.-B. & Praly L. (1992). Adaptive Nonlinear Regulation: Estimation from the Lyapunov Equation, *IEEE Transactions On Automatic Control*, vol. 31, No. 6, June 1992, 729-740, ISSN: 1558-0865.

Smallwood D.A. & Whitcomb, L.L. (2003). Adaptive identification of dynamically positioned underwater robotic vehicles. *IEEE Trans. on Control Systems Technology*, Vol.11, No. 4, July 2003 , 505-515, ISSN: 1558-0865.

Smallwood, D.A. & Whitcomb, L.L. (2004). Model-based dynamic positioning of underwater robotic vehicles: theory and experiment. *IEEE Journal of Oceanic Engineering*, Vol. 29, No. 1, Jan. 2004, 169-186, ISSN: 0364-9059.

Vidyasagar, M. (1993). *Nonlinear Systems Analysis*, Prentice-Hall, ISBN:0-13-623463-1, Upper Saddle River, NJ, USA.

Wang, J.-S. & Lee, C.S.G. (2003). Self-adaptive recurrent neuro-fuzzy control of an autonomous underwater vehicle. *IEEE Trans. on Robotics and Automation*, Vol. 19, No. 2, April 2003, 283-295, ISSN: 1042-296X.

Whitcomb, L.L. & Yoerger, D.R. (1999). Development, comparison, and preliminary experimental validation of nonlinear dynamic thruster models, *IEEE Journal Of Oceanic Engineering*, Vol. 24, No. 4, Oct. 1999 , 481–494. ISSN: 0364-9059.

15

An Autonomous Navigation System for Unmanned Underwater Vehicle

Seong-Gon Kim[1] and Yong-Gi Kim[2]
[1]*University of Florida*
[2]*Gyeongsang National University*
1USA
[2]*Korea*

1. Introduction

Autonomous underwater vehicles should possess intelligent control software that performs intellectual functions such as cognition, decision and action, which originally belong to the ability of domain expert, since the unmanned underwater robot is required to navigate in hazardous environments where humans do not have direct access to. In this paper, we suggest an intelligent system architecture called the RVC model, which can be applied to various kinds of unmanned vehicles. The architecture consists of the collision avoidance system, the navigation system, and the collision-risk computation system. The RVC architecture is devised to make use of artificial intelligence techniques, and to provide the subsystems structural and functional independency.

The collision avoidance system adopts a new heuristic search technique for the autonomous underwater vehicles equipped with obstacle avoidance sonar. The fuzzy relation product between the sonar sections and the properties of real-time environment is used to decide the direction for the vehicle to proceed. The simulation result leads to the conclusion that the heuristic search technique enables the AUV to navigate safely through obstacles and reach its destination goal with the optimal path. The navigation system executes the offline global path planning for the AUV to guarantee the safe and efficient navigation from its start point to the target destination. The system also does the duty of monitoring and controlling the vehicle to navigate following the directed path to destination goal. The collision-risk computation system produces a degree of collision risk for the underwater vehicle against surrounding obstacles using information from the circumstances, obstacles, and positions. The degree is provided to the collision avoidance system as one of the decision tools used for safe avoidance with the obstacles. A 3D simulator is developed to test the AUV navigation system based on the RVC model. The goal of the simulator is to serve as a testing ground for the new technologies and to facilitate the eventual transfer of these technologies to real world applications. The simulation system consists of an environment manager, objects and a 3D viewer. Objects model all physical elements such as the map, obstacles and the AUV. The environment manager plays the role of an intermediary, which allows created objects to interact with each other, and transmits information of the objects to the 3D viewer. The 3D viewer analyzes the received information and visualizes it with 3D graphics by using OpenGL primitives.

2. Intelligent system architecture

The navigation system for autonomous underwater vehicles needs various techniques to be effectively implemented. The autonomous technique usually contains complicated and uncertain factors and thus makes use of some artificial intelligence methods to solve the problems. Artificial intelligence techniques are classified largely into two categories. One is the symbolic AI technique, such as knowledge-based system, which operates in ways similar to the human thought process, and the other is the behaviour-based AI technique such as neural network or fuzzy which behaves much like human sensorial responses. The former is considered a higher-level intelligence but it alone is not enough to make a system conduct intelligently in domains where very sophisticated behaviours are needed.

2.1 RVC intelligent system model

Research in autonomous navigation systems became very active with the rapid advancement of hardware technologies during the end of the 20th century. Researchers had tried to implement intelligent control for autonomous navigations using symbolic AI techniques but they could not succeed because of the difference in representation methods between the symbolic AI techniques they were attempting to use and the actual information needed to operate the navigation system. The symbolic AI technique is adequate for problems which are well-defined and easy to represent but not for real world problems which are usually ill-defined and in most cases have no limitation. These difficulties made researchers work on the development of AI techniques that were good for solving real world problems. Reactive planning (Agre et al., 1987), computational neuroethology (Cliff, 1991), and task-oriented subsumtion architecture (Brooks, 1986) are the results of the research, and are called behaviour-based AI (Turner et al., 1993). Many researches concluded that symbolic AI or behaviour-based AI techniques alone cannot reach the allowable goal for the navigation system of unmanned underwater vehicles (Arkin, 1989) and recent researches on autonomous navigations are focused on using both AI techniques and improving the performance of the system (Arkin, 1989; Turner, 1993; Scerri & Reed, 1999; Lee et al. 2004; Bui & Kim, 2006). The two AI techniques have different characteristics and thus is hard to combine the two techniques into a single system effectively. In this article, an intelligent system model, called the RVC (Reactive Layer-Virtual World-Considerate Layer), is introduced for the effective combination of symbolic and behaviour-based AI techniques into a system.

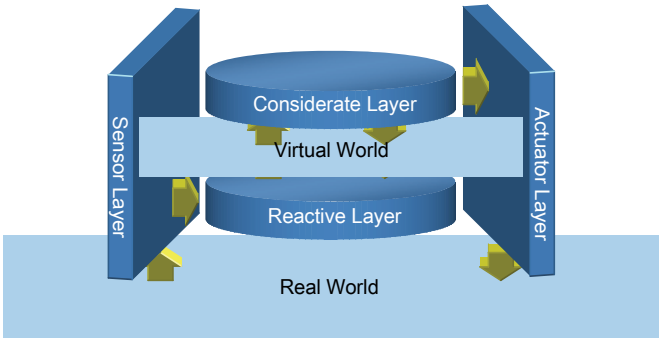


Fig. 1. RVC intelligent system

Fig.1 is the schematised RVC intelligent system model. The model is conceptualised for cordial combination of the two different AI techniques, and it also enhances the structural and functional independency of each subsystem, such as collision avoidance system, navigation system, or collision risk computation system. In this model, the reactive layer processes the uncertain problems in the real world and then passes the symbolized results to the considerate layer where the symbolic AI technique makes use of the information for the final decision. For this procedure, the model needs a common information storage space, where the information produced from the reactive layer is represented in real-time before it is consumed by the considerate layer. From the considerate layer's point of view, the information storage space resembles a subset of real world, and thus this storage space will be referred to as a 'Virtual world' henceforth.

2.2 Autonomous navigation architecture based on RVC intelligent system model

Autonomous navigation system based on the RVC intelligent system model uses the concept of information production/consumption and client/server for transferring the collected information from the real world to each module of the system in real-time. For this purpose, the intelligent navigation system contains functions such as memory management, data communication, and scheduling. Data communication in the system adopts the TCP/IP protocol, and this makes the system platform-independent and thus makes load balancing smooth. The scheduling function synchronizes the exchanging of real-time data among the modules, and it also processes possible errors in the system. The RVC intelligent system model guarantees independency among the modules in the system, and this enables the parallel development of each system module. Fig. 2 is the autonomous navigation architecture based on the RVC system model.

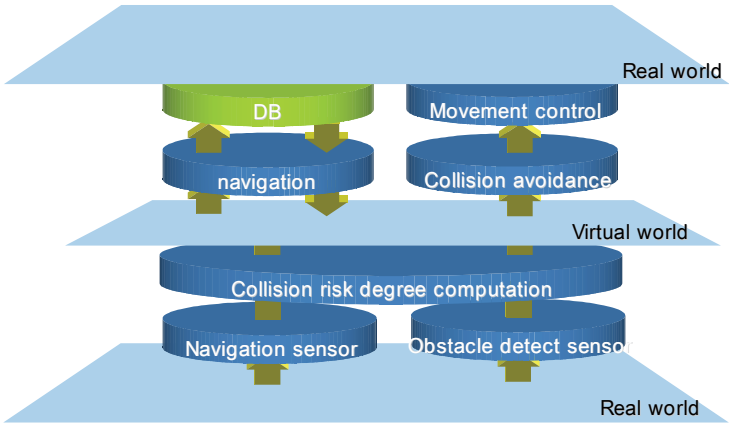


Fig. 2. Autonomous navigation architecture based on RVC system model

3. Subsystems for autonomous navigation system

3.1 Collision avoidance system

Relational representation of knowledge makes it possible to perform all the computations and decision making in a uniform relational way, by mean of special relational compositions called triangle and square products. These were first introduced by Bandler and Kohout and

are referred to as the BK-products in the literature. Their theory and applications have made substantial progress since then (Bandler & Kohout, 1980a, 1980b; Kohout & Kim, 1998, 2002; Kohout et al., 1984).

There are different ways to define the composition of two fuzzy relations. The most popular extension of the classical circular composition to the fuzzy case is so called max-min composition (Kohout et al., 1984). Bandler and Kohout extended the classical circular products to BK-products as sub-triangle (◁, "included in"), super-triangle (▷, "includes"), and square (□, "are exactly the same"). Assume the relations R and S are fuzzy relations, then the *R-afterset* of x, xR and the *S-foreset* of z, Sz, obviously are fuzzy sets in Y. The common definition of inclusion of the fuzzy set xR in Y in the fuzzy set Sz in Y is given by (1).

$$xR \subseteq Sz \Leftrightarrow (\forall y \in Y)(xR(y) \le Sz(y)) \tag{1}$$

A fuzzy implication is modeled by means of a fuzzy implication operator. A wide variety of fuzzy implication operators have been proposed, and their properties have been analyzed in detail (Bandler & Kohout, 1980c; Lee et al., 2002). For this study, we make use only of operator 5 as shown in (2).

$$a \rightarrow_5 b = \min(1, 1 - a + b) \tag{2}$$

Using (2), with n the cardinality of Y, we easily obtain the definitions for the sub-triangle and supper-triangle products in (3), (4) while the square product using the intersection and the minimum operator is shown in (5) and (6) respectively.

$$x_i(R \triangleleft S)z_j = \frac{1}{n}\sum_{y \in Y} \min(1, 1 - x_i R(y) + Sz_j(y)) \tag{3}$$

$$x_i(R \triangleright S)z_j = \frac{1}{n}\sum_{y \in Y} \min(1, 1 + x_i R(y) - Sz_j(y)) \tag{4}$$

$$x_i(R \square S)z_j = x_i(R \triangleleft S)z_j \cap x_i(R \triangleright S)z_j \tag{5}$$

$$x_i(R \square S)z_j = \min(x_i(R \triangleleft S)z_j, x_i(R \triangleright S)z_j) \tag{6}$$

Along with the above definitions, α-cut and Hasse diagram are also the two important features of this method. The *α-cut* transforms a fuzzy relation into a crisp relation, which is represented as a matrix (Kohout & Kim, 2002; Kohout et al., 1984). Let R denotes a fuzzy relation on the $X \times Y$, the *α-cut* relation of R is defined as the equation (7).

$$R_\alpha = \{(x, y) \mid R(x, y) \ge \alpha \text{ and } 0 \le \alpha \le 1 \tag{7}$$

The Hasse diagram is a useful tool, which completely describes the partial order among the elements of the crisp relational matrix by a Hasse diagram structure. To determine the Hasse diagram of a relation, the following three steps should be adopted (Lee & Kim, 2001).

Step 1. Delete all edges that have reflexive property.

Step 2. Eliminate all edges that are implied by the transitive property.

Step 3. Draw the diagraph of a partial order with all edges pointing upward, and then omit arrows from the edges.

In this study it is required that obstacle avoidance sonar range can be partitioned into several sub-ranges. One of these represents for the successive heading candidate for AUVs to go ahead. Whenever obstacle is detected, the sonar return is clustered and the sections in which obstacles present can be identified. The sonar model is illustrated as in Fig.3. Domain experts who have wide knowledge about ocean science could give the properties about the environmental effects to the of AUVs navigation.

A forward looking obstacle avoidance sonar whose coverage range can be divided into multi-sections is used to determine a heading candidate set *S*. Otherwise, a property set *P* describes the effects of AUVs toward the real time environment. The fuzzy rule base and membership function for the corresponding property can be estimated subjectively by the expert knowledge. With the set of the candidate $S = \{s_1, s_2, s_3, ..., s_i\}$ and the set of environmental properties $P = \{p_1, p_2, ..., p_j\}$, the relation *R* is built as (8). The elements r_{ij} of this relation mean the possibility the section s_i can be characterized by the property p_j. The value of r_{ij} is calculated by means of the rule bases with the membership functions.

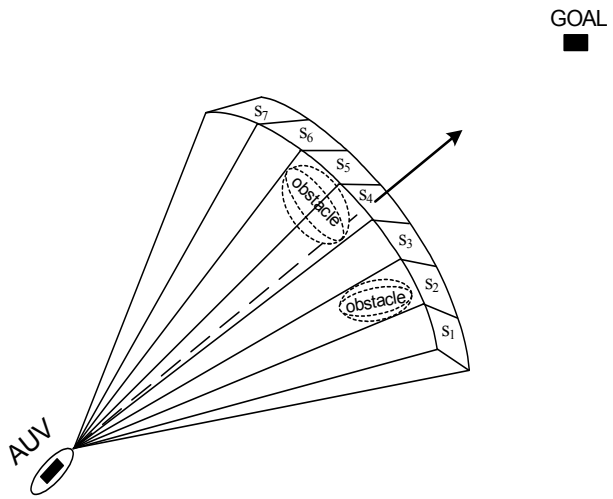


Fig. 3. A model of forward looking obstacle avoidance sonar

$$R = S \times P = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1j} \\ r_{21} & r_{22} & \cdots & r_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ r_{i1} & r_{i2} & \cdots & r_{ij} \end{bmatrix} \begin{matrix} s_1 \\ s_2 \\ \vdots \\ s_i \end{matrix} \qquad \begin{matrix} p_1 & p_2 & \cdots & p_j \end{matrix} \tag{8}$$

$$T = R \triangleleft R^T = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1i} \\ t_{21} & t_{22} & \cdots & t_{2i} \\ \vdots & \vdots & \ddots & \vdots \\ t_{i1} & t_{i2} & \cdots & t_{ii} \end{bmatrix} \begin{matrix} s_1 \\ s_2 \\ \vdots \\ s_i \end{matrix} \qquad \begin{matrix} s_1 & s_2 & \cdots & s_i \end{matrix} \tag{9}$$

$$R_\alpha = \alpha\_cut(T,\alpha) = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1i} \\ a_{21} & a_{22} & \cdots & a_{2i} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ii} \end{bmatrix} \begin{matrix} s_1 \\ s_2 \\ \vdots \\ s_i \end{matrix} \qquad (10)$$
$$\begin{matrix} s_1 & s_2 & \cdots & s_i \end{matrix}$$

In the next step a new fuzzy relation T is computed by using sub-triangle product $\triangleleft$ to fuzzy relation R and R^T, the transposed relation of R. The fuzzy relation T as shown in (9) is the product relation between candidate set S that means the degree of implication among elements of candidate set. Then, the *α-cut* is applied to fuzzy relation T in order to transform into crisp relation as shown in (10). It is important to select a reasonable *α-cut* value because the hierarchical structure of candidate set depends on an applied *α-cut*. Finally, we draw the Hasse diagram, which completely describes a partial order among elements of candidate set, that is to say, a hierarchical structure among the elements of candidate set with respect to the optimality and efficiency. Select then the top node of the Hasse diagram as the successive heading direction of AUVs.

Because the energy consumption in vertical movement of AUVs is much greater than in the horizontal movement (1.2 times) (Ong, 1990), this technique focus strongly on the horizontal movement. In the case of obstacle occurrence, AUVs just turn left or turn right with the turning angle determined by degree from the current heading to the selected section. But in the exception case a very wide obstacle has completely filled up the sonar's coverage, AUVs must go to up one layer at a time and then apply the algorithm to find out the turning. Until obstacle clearance, AUVs are constrained to go back to the standard depth of the planned route.

The algorithm of the proposed technique can summarize into five below steps and is imitated briefly in control flow as shown in Fig. 4.

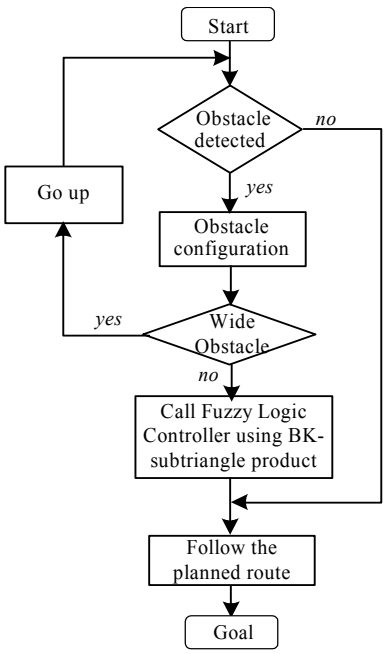


Fig. 4. A control flow of collision avoidance of AUV

Step 1. If AUVs detects obstacle then go to next step, else go to step 5
Step 2. Determine P and configure S
Step 3. If very wide obstacle is detected in all of S then go up and return step 1; else go to next step.
Step 4. Call the fuzzy logic controller using BK-subtriangle product to S and P to figure out the successive heading for obstacle avoidance
Step 5. Go on in the planned route

3.2 Navigation

Generally, the navigation system of unmanned underwater vehicles consists of two fucntions. One is path planning and the other is guidance and control (Vasudevan & Ganesan, 1996; Oommen et al., 1987). Path planing is the fuction of setting a path from a start point to a target destination using waypoints, and the function of guidance and controlling is to monitor and guide the vehicle to follow the designated path. The duty of the navigation system of the unmanned underwater vehicle in this article is transferring the following information into the autonomous navigation system's Virtual world: first, the results of an offline global path planning which allows the system a safe and optimal operation from start point to target destination, and secondly, monitoring and controlling the vehicle to stay on the set path to target destination. Fig. 5 shows the stucture of the navigation system

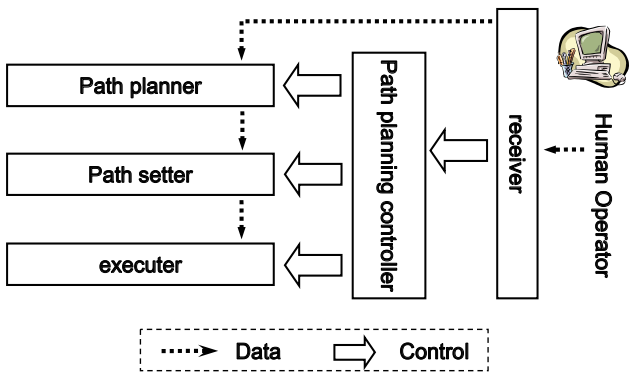


Fig. 5. Navigation system

Unmanned underwater vehilcles operate in a 3-dimensional environment and the vehicles do not have to consider static obstacles that are located below a certain depth. Global path planning for the autonomous navigation system adopts a new palnning algorithm (Kim, 2005) in which points of contact with the obstacles and waypoint trees are utilized to get the optimal path to the target destination. To get the global path, this algotithm computes the position of contact points between the start point and the static obstacle, and then connects the contact points to produce a waypoint tree. The waypoint tree is searched using a depth-first search algorithm to get the optimal path to the destination. The waypoints produced are delievered to the Virtual world, and will be used by other subsystems such as the collision avoidance system.

Fig. 6 shows an example of paths produced using the contact points when there is a static obstacle between the start point and the destination. First, it calculates the position of the left

contact point Ls and the right contact point Rs between the start point S and the obstacle, then it calculates the position of the left contact point Lg and the right contact point Rg between the destination G and the obstacle. Then, the contact points between Ls and Lg and the contact points between Rs and Rg are calculated recursively. The produced paths and contact points are stored using the data structure shown in Fig. 7 where the coordinates of the points are the data of the node, and the pointers are directed to next nodes.

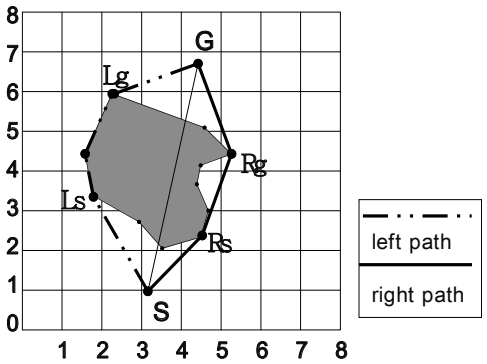


Fig. 6. Path planning

Fig. 7. The structure of node

When more than one obstacle exists between the start point and the target destination, the algorithm produces a waypoint tree for each contact point of the obstacle. Fig. 8 is a marine chart of such case, and the waypoint tree is shown in Fig. 9. With the waypoint tree, one can extract the obstacles that actually affect navigation of the vehicle from all the static obstacles existing between start point S to destination point G. Information of the left and right paths for avoiding the obstacles will be stored in the waypoint tree. The waypoint tree will have the minimum required information for producing all the paths from start point S to destination point G.

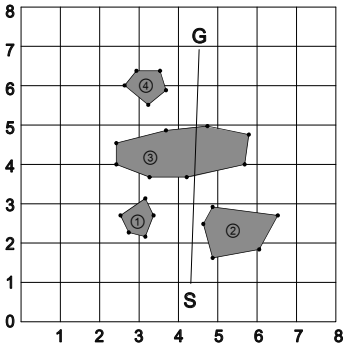


Fig. 8. A marine chart with multiple obstacles

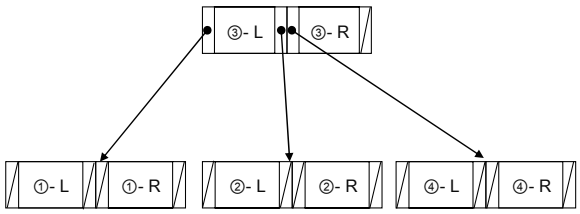


Fig. 9. Way-point tree

3.3 Collision risk computation system

The Collision Risk Computation System uses information from the surrounding environment as well as the obstacle and positioning information to compute the risk of the autonomous underwater vehicle colliding with various obstacles that exist in its environment (Kim, 2001; Hara & Hammer, 1993). The system provides a basis for the decisions it makes so that if the system finds the autonomous underwater vehicle at risk of colliding with an obstacle, it changes the navigation path so that it can safely avoid the obstacle.

The Collision Risk Computation System uses fuzzy inference which consists largely of 3 modules as seen in Fig. 10 to compute collision risks the autonomous underwater vehicle might face while navigating in its environment. The first module is the input module that reads in the vector information of the autonomous underwater vehicle and obstacle from the Virtual world, then computes the obstacle's DCPA(Distance of the Closest Point of Approach) and TCPA(Time of the Closest Point of Approach). The second Collision Risk Computation Module then uses fuzzy logic to calculate the risk of collision. It fuzzifies the DCPA and TCPA from the first module and performs a fuzzy-inference, then defuzzifies it to compute the risk of collision. In order to send the computed collision risk value to the Collision Risk Computation System, the third Output Module takes the computed collision risk and transfers it to the Virtual world.

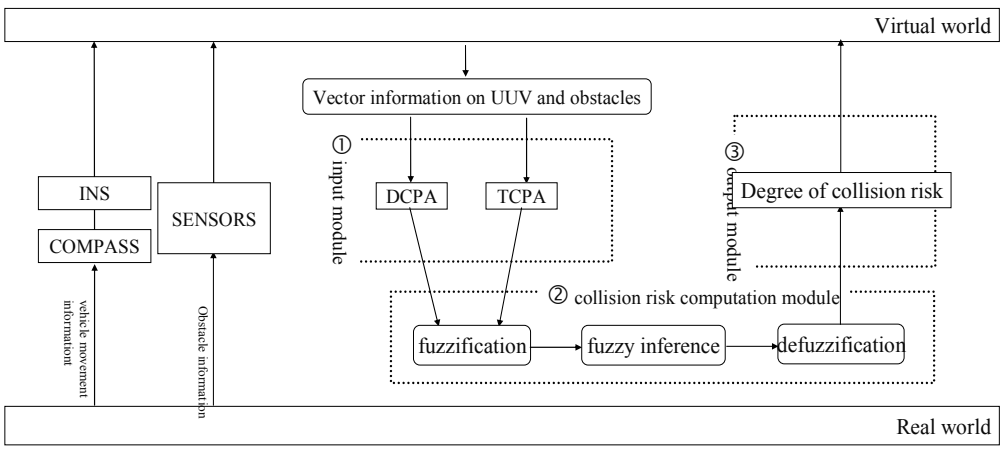


Fig. 10. Collision-risk computation system

The collision risk is computed by fuzzy-inference using DCPA and TCPA as its input. The inference rule uses the centroid method with the min operation as the antecedent and the

product operation as the consequent. The membership functions of DCPA and TCPA, which are the input values, and the collision risk, which is the output value, are first defined. Fig. 11, Fig. 12, and Fig. 13 show the membership functions of the DCPA, TCPA and collision risk, respectively. The labels used for each membership function is as follows:

P : Positive, N : Negative, S : Small, M : Medium, B : Big

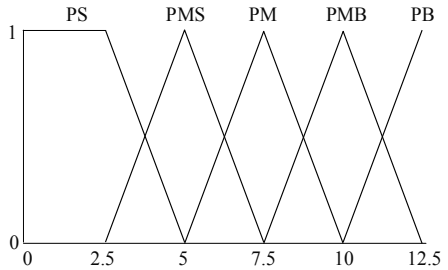


Fig. 11. Membership function of DCPA(meter)

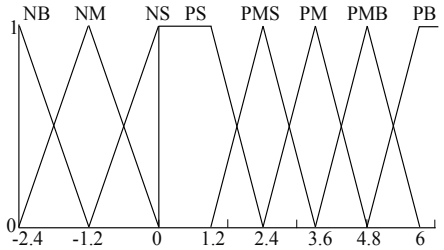


Fig. 12. Membership function of TCPA(second)

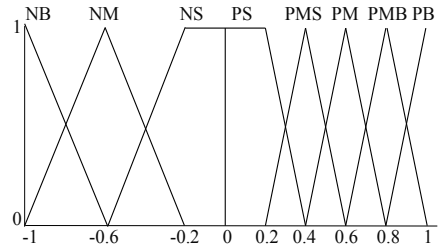


Fig. 13. Membership function of degree of collision risk

TCPA / DCPA	NB	NM	NS	PS	PMS	PM	PMB	PB
PS	NS	NM	NB	PB	PMB	PM	PMS	PS
PMS	NS	NS	NM	PMB	PM	PMS	PS	PS
PM	NS	NS	NS	PM	PMS	PS	PS	PS
PMB	NS	NS	NS	PMS	PS	PS	PS	PS
PB	NS	NS	NS	PS	PS	PS	PS	PS

Table 1. Inference rules for degree of collision risk

Table 1 is the inference rule table used to compute the collision risk. It was preset based on a navigation expert's knowledge and this can be changed depending on any particular navigator's experience or knowledge.

4. Simulation system

The simulation system is a software that lets users experiment through a computer the autonomous underwater vehicle's overall navigation process as well as the objects that appears during the navigation by modeling all the states, factors, objects, devices etc. associated with the autonomous underwater vehicle and equipping it with a virtual autonomous navigation system. The simulation system appropriately models the various sensors, the speedometer and other navigation devices that work as input, and the propulsion and steering device that work as output so that they can interact with the autonomous navigation system.

The autonomous underwater vehicle's navigation system receives information about its environment necessary for navigation as input from various sensors, then analyzes this information to send new action commands to the propulsion or steering device. The simulation system replaces navigation sensors, obstacle sensors, and movement controller in the RVC model that was presented in chapter 2, and displays the execution result on the screen. As seen in Fig. 14, the simulation system simulates the interaction between the autonomous navigation system and the physical devices, then displays the result in 3D. This system largely consists of an environment manager, objects and a 3D viewer.

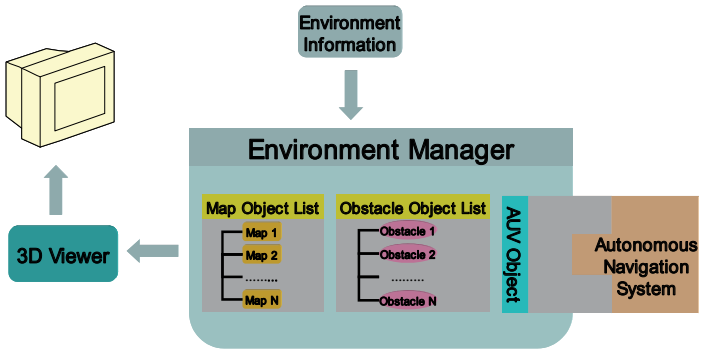


Fig. 14. Structure of simulation system

The environment manager takes environment information such as Table 2 and models the factors that influence the autonomous underwater vehicle's navigation while enabling interaction between the objects that are involved. The autonomous navigation system references the various information generated by the environment manager and carries out its tasks. The main function of the environment manager is to construct and manage the simulation environment by generating various objects and initializing its properties while tuning the interaction between these objects.

An object is the object oriented representation of the state of the environment information and the autonomous underwater vehicle. To represent the autonomous underwater vehicle's navigation environment, the simulator creates subjects of identical properties into classes, and these classes are the main frame of objects. The objects generated by the environment manager represents the navigation environment and has specific properties

and motions depending on its given function. The environment manager generates a map, obstacle and the autonomous underwater vehicle as an object. The map object represents the ocean floor geography that the autonomous underwater vehicle navigates through and has a hierarchical structure constituting of Areas and Spots. The obstacle object represents all obstacles that can appear during navigation and can be divided in to dynamic obstacles and static obstacles depending on their movement. The autonomous underwater vehicle object represents the autonomous underwater vehicle itself and is connected with the autonomous navigation system. Fig. 15 shows the overall environment information classes.

Subject	Start symbol	Input content	Example
Area	E	Entire area range	E200 200
UUV	A	UUVs position value	A-67.3 3.3 81.4
Obstacle	O	Obstacles position value	O-7.8 0.0 -29.4
Geography	M	Grid's relative position value Grid's four corner altitude value Spot altitude value Grid's unitvector value	M00001317.2 8.6 12.9 17.2 17.2 -0.7 0.1 0.7

Table 2. Environment Information

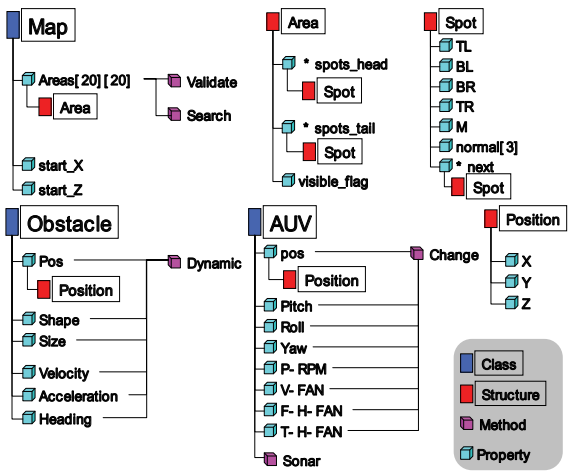
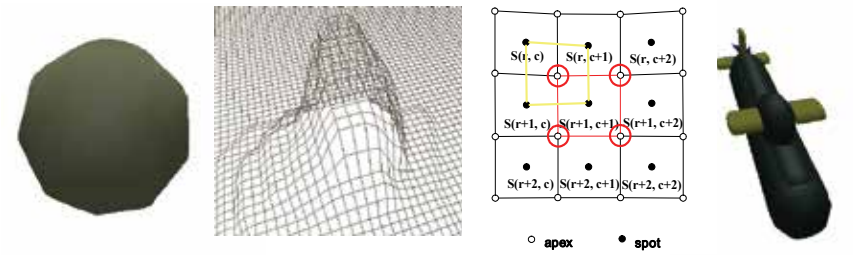


Fig. 15. Environment information class

The 3D Viewer is the component that displays the simulation process in 3D as seen in Fig. 16. The 3D viewer receives information from the environment manager as input and uses OpenGL to render the simulation process in 3D. OpenGL is the software interface to the graphical hardware allows the generation of objects or computations necessary in producing 3D applications. It can run on various hardware platforms but does not support commands with the ability to generate complex objects and can only generate primitives such as points, lines and polygons. For the representation of complex structures such as geography, obstacles and the autonomous underwater vehicle object, the primitives are combined to build the objects necessary for the simulation.



obstacle map uuv

Fig. 16. 3-D viewer for object

The Collision Avoidance System, Collision Risk Computation System and the Simulation System were used together in the simulation to test the performance of the presented underwater vehicle's autonomous navigation. For the specifications necessary for the simulation, the autonomous underwater vehicle developed by the Korean Agency for Defense Development was used and is shown in Table 3.

Spec.	Value
Vehicle length/diameter	10 (ratio)
Max speed	8.0kts
Max operation depth	100m
Displacement tonnage	1.380kg

Table 3. Specification of UUV

The underwater vehicle's autonomous navigation system was tested using a scenario where three dynamic obstacles exist. The autonomous underwater vehicle's starting point and destination point were set to S(0,0,-10) and G(0,210,-10), respectively. The first obstacles starting point and destination point were set to O(-20),150,-10) and O(13,-60-10), respectively, where it approaches the autonomous underwater vehicle from the front left side. The second obstacle approaches the autonomous underwater vehicle from the rear with the starting point and destination point set to O(30,-10,-17) and O(-2,160,-17), and the last obstacle approaches the UUV directly from the front with the starting point and destination point set to O(0,200,-10) and O(0,-20,-10), respectively. Fig. 17 shows the actual

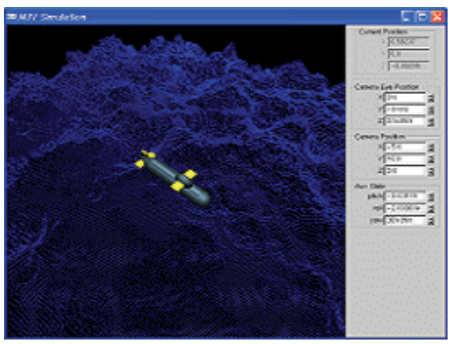
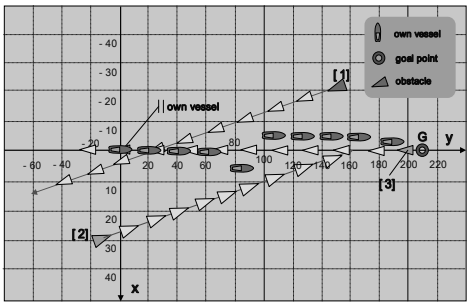
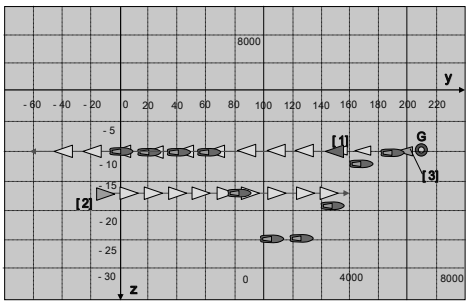


Fig. 17. Display of simulation

simulation in progress and Fig. 18 shows the results as a map to help understand the simulation results. As shown in the simulation results, the autonomous underwater vehicle detected the first approaching obstacle O(-11,87,-10) at point P(0,63,-10) and sends an avoidance command to point P(7,84,-17), then continues to avoid the second obstacle O(18, 58, -17) to point P(-7, 105, -24), and this confirmed that the collision avoidance performed reasonably and efficiently.



(a). Simulation result in view of [X-Y] axis



(b). Simulation result in view of [Y-Z] axis

Fig. 18. Simulation result with scenarios

5. Conclusion

This paper designed a RVC intelligent system model that can be applied to various unmanned vehicles and the underwater vehicle's intelligent autonomous navigation system was designed consisting of a collision avoidance system, a navigation system and a collision risk computation based on a Virtual world system. During the development of the Virtual world system, several points such as the fusion of different techniques, preservation of system consistency, real time system processing etc. were taken into consideration, and since it models a client/server structure, it also has the features of consistency, independence maximization, and load balancing. The RVC intelligent system can be applied not only to autonomous underwater vehicles, but to various autonomous robots such as unmanned aerial vehicles, mobile robots and autonomous submarines. To test the performance of the underwater vehicle's intelligent autonomous navigation system based on this RVC intelligent system model, a 3D simulator was developed, and through a scenario with dynamic obstacles existing in the navigational environment, the validity of the intelligent autonomous navigation system was verified.

6. References

Agre, E. & Chapman, D. (1987). An Implementation of a theory of activity, *Proceedings of the Sixth National Conference on Artificial Intelligence*

Arkin, R. (1989). Towards the Unification of Navigational Planning and Reactive Control, *Proceeding of the AAAI Spring Symposium on Robot Navigation*, Mar. 1989

Bandler, W. & Kohout, L (1980a). Fuzzy Relational Products as a Tool for Analysis and Synthesis of the Behavior of Complex natural and Artificial System, In: *Fuzzy Sets: Theory and Application to Analysis and Information Systems*, Wang, K, & Chang, P. (Ed.)., (341-367), Plenum Press, New York

Bandler, W. & Kohout, L. (1980b). Semantics of Implication Operators and Fuzzy Relational Products, *International Journal of Man-Machine Studies*

Bandler, W & Kohout L. (1980c). Fuzzy Power Sets and Fuzzy Implication Operator, *Fuzzy Set and System,* No. 4, (13-30)

Brooks, A. (1986), A robust layered control system for a mobile robot," IEEE Journal of Robotics and Automation

Bui, L. & Kim, Y. (2006). An Obstacle Avoidance Technique for Autonomous Underwater Vehicles Based on BK-Products of Fuzzy Relation, *Fuzzy Sets and Systems*, Vol. 157, (560-577)

Cliff, D. (1991), A provisional manifesto, *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, The MIT Press

Hara, K. & Hammer, A. (1993). A safe Way of Collision Avoidance Maneuver based on Maneuvering Standard using Fuzzy Reasoning Model, *Proceedings of MARSIM 93*, 1993, St. John's

Kim, E. (2001). Collision-risk computation system for collision avoidance. *Transaction of Korean Fuzzy and Intelligent System Association*, Vol. 2, (524-527)

Kim, Y. (2005). Autonomous ship control system using intelligence techniques, Technical Report, TECD-417, Korean Agency for Defense Development

Kohout, L.; Keravnou, E. & Bandler, W. (1984). Automatic Documentary Information Retrieval by Means of Fuzzy Relational Products, In: *Fuzzy Sets in Decision Analysis*, Gaines, B. R.; Zadeh, L. A. & Zimmermann, H. J., (Ed.), (308-404), North-Holland, Amsterdam

Kohout, L. & Kim E. (1998). Semiotic descriptors in fuzzy relational computations, Proc IEEE Int Symp Intelligent Control, IEEE Int Symp Computational Intelligence in Robotics and Autonomous and Intelligent Systems and Semiotic (A Joint Conf Science and Technology of Intelligent Systems), pp. 828-833, 1998, Piscataway

Kohout L. & Kim E. (2002). The role of BK-products of Relations in Soft Computing, *Soft Computing,* Springer-Verlag, No. 6, 92-115

Lee, Y. & Kim, Y. (2001). An Intelligent Navigation System for AUVs using Fuzzy Relational Products, *IFSA World Congress and 20th NAFIPS International Conference, Joint 9th*, Vol.2 , 709-714

Lee, Y.; Kim, Y. & Kohout, L. (2004). An Intelligent Collision Avoidance System for AUVs using Fuzzy Relational Products, *Information Sciences*, Vol. 158, 209-232

Lee, Y.; Noe, C. & Kim, Y. (2002).Implication Operators in Fuzzy Relational Products for a Local Path-Planning of AUVs, *Proceedings of NAFIPS, Annual Meeting of the North American,* Fuzzy Information Processing Society, 22

Ong, M. (1990). A Mission Planning knowledge-based system with Three-Dimensional Path Optimization for the NPS Model 2 Autonomous Underwater Vehicle, Master's Thesis, Naval Postgraduate School

Oommen, B. et al. (1987). Robot Navigation in Unknown Terrains Using Learned Visibility Graphs. Part I: The Disjoint Convex Obstacle case, *IEEE journal of Robotics and Automation*, Vol. RA-3, No. 6, (672-681)

Scerri, P. & Reed, N. (1999). Requirements for a General Agent Architecture for Agent-Based Simulation Environments, *Agents'99 Workshop on Autonomy Control Software*, May, 1999

Turner, R.; Blidberg, R.; Chappell, S. & Jalbert, J. (1993). Generic behaviors: An approach to modularity in intelligent systems control, *Proceedings of the 8th International Symposium on Unmanned Untethered Submersible Technology*, 1993

Vasudevan, C. & Ganesan, K. (1996). Case-Based path Planning for Autonomous Underwater Vehicles, In: *Underwater Robots*, Yuh, Ura and Bekey (Eds.), (1-15), Kluwer Academic Publishers, Boston

16

Schooling for Multiple Underactuated AUVs

Ji-Hong Li, Bong-Huan Jun, Pan-Mook Lee and Yong-Kon Lim
Maritime & Ocean Engineering Research Institute, KORDI,
Republic of Korea

1. Introduction

In the past few decades, autonomous underwater vehicles (AUVs) have been playing one of most important roles in the applications ranging from scientific research, survey to industry and military operations. Today, there is an apparent trend that more and more underwater tasks are carrying out by cooperative operations of multiple AUVs instead of traditional method of using single AUV (Soura & Pereira, 2002; Edwards et al., 2004; Guo et al., 2004; Watanabe & Nakamura, 2005; Fiorelli et al., 2006). Multiple AUVs have cost-effective potential. However, a number of research efforts are still remained to be done before this advanced technology can be fully applied in the practice. And one of the efforts is about the efficient schooling scheme for these multiple underwater vehicles.

The history of the formation or cooperative control of multiple agent systems can be traced back to the 1980's. Reynolds (1987) introduced a distributed behavioural model for flocks of birds, herds of land animals, and schools of fishes. This model can be summarized as three heuristic rules: flock centring, collision avoidance and velocity matching. In the formation algorithm (Reynolds, 1987), each dynamic agent was modelled as certain particle system – a simple double-integrator system. This kind of agent model has been inherited in most of the following research works (Leonard & Fiorelli, 2001; Olfati-Saber & Murray, 2002, 2003; Fiorelli et al., 2006; Olfati-Saber, 2006; Do, 2007). Besides these works, another type of linear model was used in Smith et al. (2001), and certain nonlinear model was applied for underwater vehicles (Dunbar & Murray, 2002) and for wheel robots with terminal constraints (Fax & Murray, 2004). In both of Dunbar & Murray (2002) and Fax & Murray (2004), the nonlinear dynamics were all fully actuated.

In this chapter, we consider the schooling problem for multiple underactuated AUVs, where only three control inputs - surge force, stern plane and rudder are available for each vehicle's six degrees of freedom (DOF) motion. For these torpedo-type underwater flying vehicles, since there are non-integrable constraints in the acceleration dynamics, the vehicles do not satisfy Brockett's necessary condition (Brockett et al., 1983), and therefore, could not be asymptotically stabilizable to an equilibrium point using conventional time-invariant continuous feedback laws (Reyhanoglu, 1997; Bacciotti & Rosier, 2005). Moreover, these vehicles' models are not transformable into a drift-less chained form (Murray & Sastry, 1993), so the tracking method proposed in Jiang & Nijmeiner (1999) cannot be directly applicable to these vehicles. Recently, quite a number of research works have been carried out on the tracking of underactuated surface ships (Jiang, 2002; Do et al., 2002a, 2002b, 2004, 2005; Pettersen & Nijmeijer, 2001; Fredriksen & Pettersen, 2006). However, the presented

tracking methods were all case-by-case that strongly depended on the ship's specifically simplified dynamics. Therefore, these tracking methods also cannot be directly applicable to the case of underwater vehicles. Since the sway and heave forces are unavailable, the most challenge in the tracking control is how to properly handle the vehicles' sway and heave dynamics in the position tracking. To deal with this problem, in this chapter, we introduce a certain polar coordinates transformation for the vehicle's velocities in the body-fixed frame. Through this coordinates transformation, each vehicle's dynamics can be transformed to a certain two inputs nonlinear strict-feedback form, according to which the proposed schooling scheme is derived.

For the torpedo-type underwater flying vehicles considered in this chapter, the pitch and yaw moments are proportional to the square of the vehicle's forward speed. From this point of view, the pitch and yaw moments are not exactly independent with the surge force. If the vehicle's forward speed is taken small value, then the pitch and yaw moments are also have to take small values, therefore, in this case we cannot fully excite the vehicle's pitch and yaw dynamics. To appropriately taking these three only available control inputs – surge force, pitch and yaw moments as independent ones, the vehicle's forward speed has to be guaranteed to take considerable magnitude. For this reason, in this chapter, firstly we assume that the vehicle's forward speed satisfies the above consideration. And the proposed schooling scheme, which is derived under this assumption, reversely can guarantee the assumption always to be fulfilled under certain initial conditions.

The common method of formation among the schemes presented so far is to apply certain potential function to conduct the agents' group behaviour. The potential function initially used in the robotics for mobile robot's motion planning (Latombe, 1991; Rimon & Koditschek, 1992), and recently widely applied in the formation of multiple agents systems (Leonard & Fiorelli, 2001; Olfati-Saber, 2006; Do, 2007). Aforementioned Reynolds's three heuristic rules of flock centring, collision avoidance, and velocity matching, which are also known as cohesion, separation, and alignment, are usually embodied by suitably selected potential functions. In Leonard & Fiorelli (2001), only 1 time differentiable function was used as potential for group formation, while $p(p \geq 2)$ times differentiable one was applied in Do (2007) and a specific smooth potential was used in Olfati-Saber (2006). In this chapter, general form of smooth potential function is introduced and similar to Olfati-Saber (2006), the potential consists of three different components: one is for the interactions between vehicles, another is for group navigation, and the third is for obstacle avoidance. Unlike aforementioned previous works, in this chapter, the vehicle's orientation is also considered. Therefore, we have to discuss the vehicles orientation matching as well as their velocity matching. Proposed schooling scheme guarantees local minimum of the vehicles formation, and the group's velocity and orientation matching in terms of polar coordinates, while keeping obstacle avoidance.

The remainder of this chapter is organized as follows. In Section 2, the vehicles' kinematics and dynamics in the Cartesian frame are presented. Through certain polar coordinates transformation, the vehicle's model can be transformed to certain two inputs strict-feedback form. Proposed vehicles schooling rules are discussed in Section 3, and corresponding formation control laws are derived in Section 4. To illustrate the effectiveness of proposed schooling scheme, some numerical simulations are carried out and analyzed in Section 5. Finally, a brief summary and some of future works are discussed in Section 6.

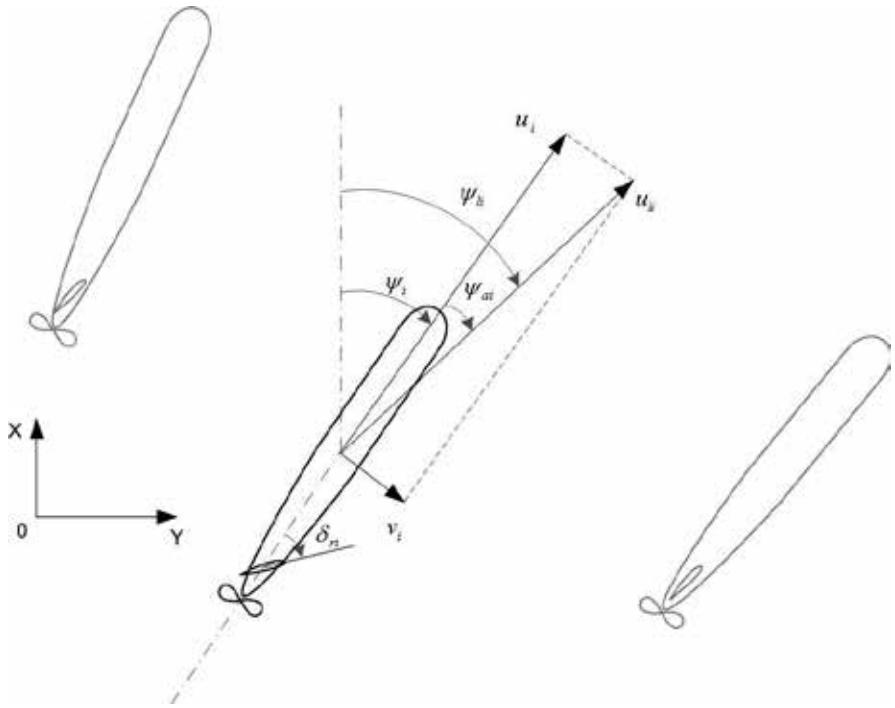


Fig. 1. General framework for an AUV in the horizontal plane.

2. Problem formulation and preliminaries

Consider a group of torpedo-type AUVs, where only surge force and yaw moment are available for each vehicle's three DOF horizontal motion[1], see Fig. 1. To date over 400 true AUVs have been built (Westwood et al., 2007) and most of them such as REMUS AUVs (Prestero, 2001) and HUGIN AUVs (Marthiniussen et al., 2004) have this type of mechanical structure. For this kind of underactuated underwater vehicles, their horizontal kinematics and dynamics can be expressed as following (Fossen, 2002; Li & Lee, 2008)

$$\begin{aligned}\begin{bmatrix}\dot{x}_i\\ \dot{y}_i\\ \dot{\psi}_i\end{bmatrix}&=\begin{bmatrix}\cos\psi_i & -\sin\psi_i & 0\\ \sin\psi_i & \cos\psi_i & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}u_i\\ v_i\\ r_i\end{bmatrix},\\ \begin{bmatrix}\dot{u}_i\\ \dot{v}_i\\ \dot{r}_i\end{bmatrix}&=\begin{bmatrix}f_{ui}(u_i,v_i,r_i)\\ f_{vi}(u_i,v_i,r_i)\\ f_{ri}(u_i,v_i,r_i)\end{bmatrix}+\begin{bmatrix}b_{ui} & 0\\ 0 & 0\\ 0 & b_{ri}\end{bmatrix}\begin{bmatrix}\tau_{ui}\\ \tau_{ri}\end{bmatrix},\end{aligned} \tag{1}$$

where $(x_i, y_i), i=1,\cdots,n$ denotes the position of the ith vehicle and ψ_i is yaw angle, all in the earth-fixed frame; u_i, v_i and r_i denote the velocities each in the surge, sway and yaw

[1] For the convenienceof discussion, in this chapter we only consider the vehicle's three DOF motion in the horizontal plane instead of its full six DOF motion.

directions in the vehicle's body-fixed frame; $f_{ui}(\cdot), f_{vi}(\cdot), f_{ri}(\cdot) \in C^1$ are the vehicle's nonlinear dynamics including hydrodynamic damping, inertia (including added mass terms) and gravitational terms in the surge, sway and yaw directions; surge force τ_{ui} and yaw moment τ_{ri} are two available control inputs with nonzero constant gains b_{ui} and b_{ri}.

For this torpedo-type underwater flying vehicles, their yaw moments τ_{ri} is proportional to the square of the forward speed, in other word, $\tau_{ri} \propto u_i^2 \delta_{ri}$ where δ_{ri} is the vehicle's rudder angle (Fossen, 2002), see Fig. 1. From this point of view, yaw moment τ_{ri} is not exactly independent of surge force τ_{ui}. Moreover, if the vehicle's forward speed takes value too small, then the yaw moment is also forced to take small value. Therefore, in this case, we could not fully excite the vehicle's yaw dynamics. In order to appropriately taking these two control inputs τ_{ui} and τ_{ri} as independent ones, in the remainder of this chapter, we make the following assumption on each vehicle's dynamics.

Assumption 1. For each vehicle in the considering group, its dynamics satisfies the following conditions.

C1. $u_i \geq u_{\min} > 0$, where $u_{\min}$ is a design parameter.

C2. For bounded u_i and r_i, v_i is also bounded and have $|v_i| \leq v_{\max}$ with $v_{\max}$ a known positive constant.

Remark 1. At first glance, in the above assumption, the condition that $v_{\max}$ is known seems too restrictive in terms of control engineering. However, in the case of underwater vehicles, it becomes very reasonable. For underwater flying vehicles, because of the effect of hydrodynamic damping terms, which is usually proportional to the square of the vehicle's corresponding speed (Fossen, 2002; Newman, 1977), they only have to take limited magnitude of velocities under the limited thrust force. Therefore, in practice, given a torpedo-type flying vehicle, it's maximum forward and sway speeds and yaw angular velocity are all easy to bring out through certain simple experiments such as basin test. From this point of view, it is reasonable for us to design the parameter $u_{\min}$ such that $u_{\min} \geq \rho v_{\max}$.

As aforementioned, since the sway force is unavailable, the most difficulty for the control of (1) is how to properly handling the vehicle's sway dynamics. To deal with this problem, firstly we introduce a polar coordinates transformation which is defined in the vehicle's body-fixed frame as following (see Fig. 1)

$$u_{li} = \sqrt{u_i^2 + v_i^2}, \quad \psi_{li} = \psi_i + \psi_{ai}, \tag{2}$$

where $\psi_{ai} = \arctan(v_i / u_i)$ is a polar angle and also called as the sideslip angle (Fossen, 2002). Since $u_i > 0$, it is easy to verify that ψ_{ai} is defined and smooth in the domain $(-0.5\pi, 0.5\pi)$. Differentiating the first equation of (2) and further according to the relationships of $u_i = u_{li}\cos\psi_{ai}$ and $v_i = u_{li}\sin\psi_{ai}$, we can get

$$\dot{u}_{li} = \dot{u}_i \cos\psi_{ai} + \dot{v}_i \sin\psi_{ai}. \tag{3}$$

Using polar coordinates u_{li} and ψ_{li}, vehicle's kinematics can be rewritten as

$$\dot{x}_i = u_{li}\cos\psi_{li}, \quad \dot{y}_i = u_{li}\sin\psi_{li}. \tag{4}$$

For the convenience of discussion, in the remainder of this chapter, we will call u_{li} and ψ_{li} as the *i*th vehicle's velocity and heading.

Combining with (3) and (4), the vehicle's model (1) can be rewritten as following form

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\psi}_{li} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_{ai} \end{bmatrix} + \begin{bmatrix} \cos\psi_{li} & 0 \\ \sin\psi_{li} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{li} \\ r_i \end{bmatrix},$$
$$\begin{bmatrix} \dot{u}_{li} \\ \dot{r}_i \end{bmatrix} = \begin{bmatrix} f_{ui}\cos\psi_{ai} + f_{vi}\sin\psi_{ai} \\ f_{ri} \end{bmatrix} + \begin{bmatrix} b_{ui}\cos\psi_{ai} & 0 \\ 0 & b_{ri} \end{bmatrix} \begin{bmatrix} \tau_{ui} \\ \tau_{ri} \end{bmatrix}. \tag{5}$$

Since $\psi_{ai} \in (-0.5\pi, 0.5\pi)$, $\cos\psi_{ai} \neq 0$. Therefore, it is easy to see that (5) is in a form where the vehicle's sway dynamics f_{vi} can be properly handled by the surge force τ_{ui}.

3. Schooling rules

Consider a group of *n* underwater vehicles, all of which have the same model as (1). Similar to the Reynolds's (1987) three heuristic rules that are flock centring, collision avoidance and velocity matching, the schooling rules proposed in this chapter can be summarized as: *geometry of schooling*, *group navigation*, and *obstacle avoidance*. As aforementioned, the group behaviour is conducted by suitably selected potential functions.

Unlike the previous works (Leonard & Fiorelli, 2001; Olfati-Saber, 2006; Do, 2007), in this chapter we consider the following general form of smooth potential function.

Definition 1 (Smooth potential function). A scalar function $f_p(\zeta, a, b)$, where $\zeta \in [0, +\infty)$ and a and b are constants with $0 < a \leq b$, is called a smooth potential function, if it satisfies the following conditions.

C3. $f_p(\zeta, a, b)$ is smooth respect to $\zeta \in [0, +\infty)$ with monotonically decreasing at $\zeta \in [0, a)$ and monotonically increasing at $\zeta \in [a, +\infty)$.

C4. If $a < b$, then $f_p(\zeta, a, b)$ has a global minimum of zero at $\zeta = a$ with $\partial f_p / \partial \zeta = 0$ for $\forall \zeta \geq b$.

C5. If $a = b$, then $f_p(\zeta, a, b) = 0$ for $\forall \zeta \geq a$.

Remark 2. If $b < +\infty$, then $f_p(\zeta, a, b)$ is said to have a *finite cut-off* (Olfati-Saber, 2006). This kind of feature plays an important role in the group formation (Leonard & Fiorelli, 2001; Olfati-Saber, 2006; Do, 2007). On the other hand, in Olfati-Saber (2006), the potential function took a finite value when $\zeta \to 0^+$, while it took $+\infty$ in Leonard & Fiorelli (2001) and Do (2007). From a collision-free point of view, the latter one seems to have its own benefit. However, the finite value case as in Olfati-Saber (2006) may be more natural in practical flocking or schooling.

Remark 3. There are many functions satisfying *C3*~*C5*. For example,

$$f_p(\zeta, a, b) = c\int_a^{\zeta} (\tau - a)\rho(\tau / b)d\tau, \tag{6}$$

$$f_p(\zeta, a, b) = c\int_a^{\zeta} (1/\tau - a/\tau^2)\rho(\tau / b)d\tau, \tag{7}$$

where $c > 0$ is a constant and $\rho(\cdot)$ is a smooth bump function taken as following

$$\rho(\xi)=\begin{cases}1, & \xi\in[0,\,h)\\ \exp[-(\xi-h)^2/(\xi-1)^2], & \xi\in[h,1]\\ 0, & otherwise\end{cases} \tag{8}$$

where $h\in(0,1)$ is a design parameter. It is easy to verify that when $\zeta\to 0^+$, $f_p\to ca^2/2$ which is a finite value in (6) while $f_p\to+\infty$ in (7). Moreover, in (6), if $b\to+\infty$, then the potential function becomes a general quadratic form as $c(\zeta-a)^2/2$. For another example of smooth potential and bump functions, refer to Olfati-Saber (2006). Functions (6) and (7) are depicted in Fig. 2.

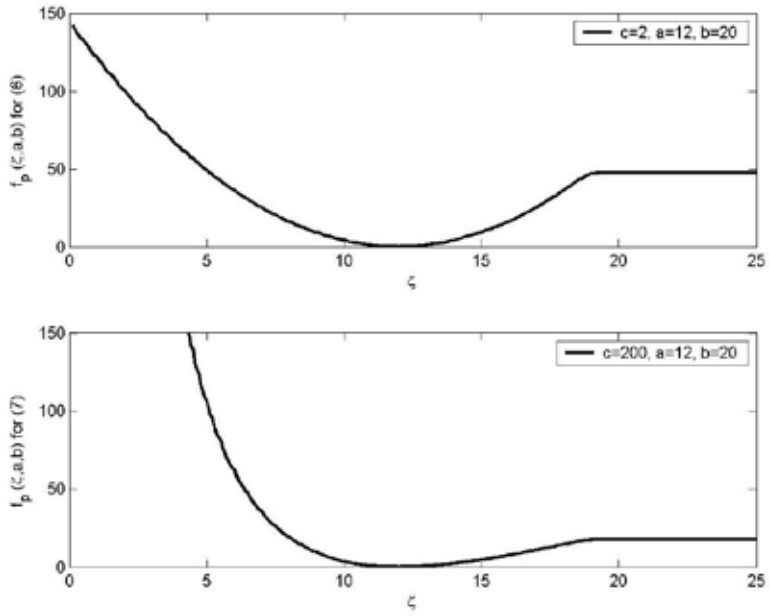


Fig. 2. Smooth potential functions for (6) and (7).

3.1 Geometry of schooling

The schooling geometry is constructed according to the following two kinds of potential functions

$$V_\alpha=\sum_{i=1}^{n}\sum_{j\neq i}^{n}f_p(\|q_j-q_i\|,a_\alpha,b_\alpha), \tag{9}$$

$$V_\beta=\sum_{i=1}^{n}\sum_{k=1}^{m}f_p(\|q_{v,k}-q_i\|,a_\beta,b_\beta), \tag{10}$$

where $q_i=(x_i,y_i)$ is the ith vehicle's coordinate and $q_{v,k}=(x_{v,k},y_{v,k})$ is the kth *virtual vehicle*'s coordinate with m the number of virtual vehicles needed to conduct the group

geometry, $a_\alpha, b_\alpha, a_\beta, b_\beta > 0$ are design parameters with $a_\alpha \le b_\alpha$ and $a_\beta \le b_\beta$, and $\|\cdot\|$ denotes the vector Euclidean norm.

The potential V_α in (9) is presenting the interactions between vehicles. From (9), it is easy to see that the desired geometry of the vehicles group is a certain net of regular triangles with vehicles located at the vertices and all side lengths are equal to a_α. If any two vehicles are apart from each other more than b_α, then there is not any cohesion between them. For (9) to be applicable, for each vehicle, all other vehicles' information including position and velocity information should be available. However, in some practical cases, this may not be available. Instead, only the information of the vehicles in its neighbour is available. In this case, the potential function can be chosen as

$$V_\alpha = \sum_{i=1}^{n} \sum_{j \in N(q_i)} f_p(\| q_j - q_i \|, a_\alpha, b_\alpha), \tag{11}$$

where $N(q_i)$ denotes the neighbour of the ith vehicle and is defined as

$$N(q_i) = \{ q_j : \| q_j - q_i \| \le b_\alpha,\ j \ne i,\ j = 1, \cdots, n \}. \tag{12}$$

In order to make V_α be smooth, in (11), it should be chosen that $a_\alpha = b_\alpha$.

In (10), the potential function V_β is for the interactions between vehicles and virtual vehicles. Here the virtual vehicles are introduced to construct the geometry of the vehicles schooling. Moreover, these virtual vehicles are used to guide the group navigation, which will be discussed in details in the next subsection. In other word, these virtual ones lead the group to follow a given desired motion. From this point of view, these virtual vehicles are also called as *virtual leaders*. Different arrangements of these virtual leaders can lead to different geometry of the schooling, see Fig. 3.

3.2 Group navigation

In the literature, group navigation is usually led by certain leaders, which can be some specific actual agents (Guo et al., 2004; Edwards et al., 2004) or some virtual ones (Leonard & Fiorelli, 2001; Olfati-Saber, 2006). The group movement can be guided through properly designing the reference paths for these virtual leaders. As aforementioned, in this chapter we apply the virtual leader concept to guide the group navigation.

Assumption 2. All virtual leaders move with the same velocity u_{lv} and same heading ψ_{lv}. Moreover, we have $u_{lv} > \sqrt{u_{\min}^2 + v_{\max}^2}$.

In fact, these virtual leaders can take any complicated motions, which in turn can lead to various geometry of the schooling. However, the focus of this chapter is taken on that to propose a *stable* schooling for a group of underactuated underwater vehicles. Therefore, for the convenience of discussion, in this chapter we only consider a simple case where all virtual leaders take the same velocity and heading. Corresponding potential function is chosen as following

$$V_{gn} = \frac{1}{2} \sum_{i=1}^{n} \left[\gamma_u (u_{lv} - u_{li})^2 + \gamma_\psi (\psi_{lv} - \psi_{li})^2 \right]. \tag{13}$$

where $\gamma_u, \gamma_\psi > 0$ are weighting factors.

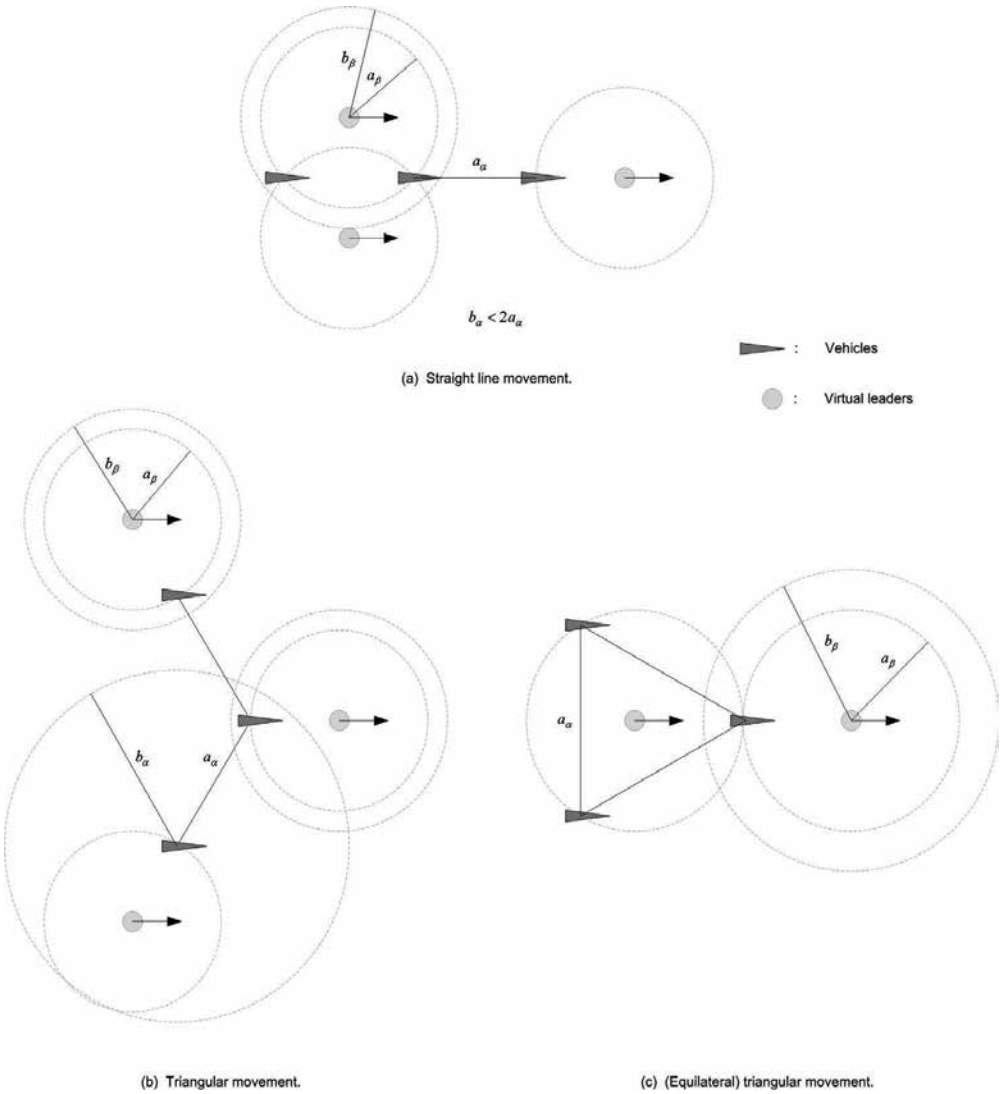


Fig. 3. Different arrangements of virtual leaders lead to different geometry of schooling.

From (13), it is easy to see that the purpose of potential V_{gn} is to force the group to keep velocity and heading matching in terms of u_{li} and ψ_{li}.

3.3 Obstacle avoidance

All obstacles considered in this chapter are position fixed. Modelling of these obstacles is as Fig. 4. Inside the circle centred at the *i*th vehicle q_i with radius b_γ, each obstacle block is modelled as the point from which to q_i is the shortest. In Fig. 4, *B1* and *B2* which are two parts of the same obstacle *B* are considered to be independent and modelled as two different points $q_{i,2}$ and $q_{i,3}$. Also, it is notable that the same obstacle such as *B* in Fig. 4 can be modelled as different points according to different vehicles.

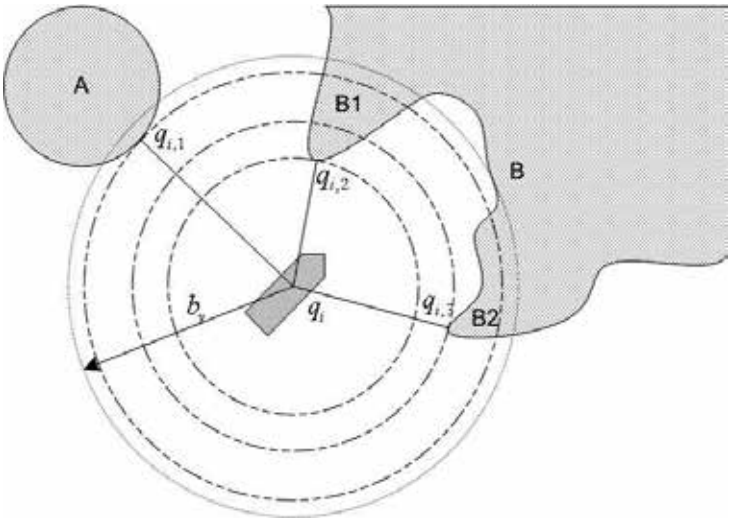


Fig. 4. Obstacle modelling.

Vehicle's obstacle avoidance is also guided by the following potential functions

$$V_\gamma = \sum_{i=1}^{n} \sum_{p \in \Omega(q_i)} f_p(\| q_{i,p} - q_i \|, a_\gamma, b_\gamma), \tag{14}$$

where $\Omega(q_i)$ is a subgroup of obstacle points defined as following

$$\Omega(q_i) = \{q_{i,p} : \| q_{i,p} - q_i \| \leq b_\gamma,\ p = 1, 2, \cdots\}. \tag{15}$$

From obstacle avoidance point of view, there is not any need to have cohesion between vehicles and obstacles. For this reason, in (14), we design the parameters such that $a_\gamma = b_\gamma$.
As aforementioned, in this chapter the group navigation is guided by the virtual leaders. Therefore, the vehicles' obstacle avoidances also have to be strongly related to the motion of virtual leaders. For this reason, we make the following assumption on the motion of the virtual leaders.

Assumption 3. All virtual leaders are designed to satisfying the following conditions.

C6. For any given obstacle, after a certain period of time, all virtual leaders always move away from this obstacle.

C7. After a period of time such that $\min_{k,i} \| q_{v,k} - q_{i,p} \| \geq b_\beta + b_\gamma$, if one or more vehicles are still trapped by obstacles, in other word, $\forall k$, $\| q_{v,k} - q_j \| \geq b_\beta$ with q_j the trapped vehicle, then all virtual leaders (smoothly) stop movement so that $\dot{q}_{v,k} = 0$. Otherwise, *C6* still satisfies.

Remark 4. From *C7* in *Assumption 3*, we can see that the obstacle avoidance scenario introduced in this chapter cannot guarantee all vehicles to flee from any given obstacles. Instead, we only try to guarantee the vehicles to collision-free with obstacles. For example, as in *C7*, if one or more vehicles are trapped by obstacles, then we force these vehicle to stop movement so as to avoid collision with obstacles. In fact, obstacle avoidance is still being a complicated and open issue in the practical robotics. How to guide the group to move through the obstacles is out of the scope of this chapter.

4. Formation control design

In this section, we will propose a formation control algorithm that can guarantee the stable schooling for multiple underactuated underwater vehicles whose models can be expressed as (1) or (5) with *Assumption 1*. It is easy to see that (5) is in a second-order nonlinear strict-feedback form, and we will solve the schooling problem using general backstepping method (Krstic et al., 1995).

Step 1. As aforementioned, the vehicles' schooling is conducted by the various potential functions introduced in the previous section. For this reason, in this step, we consider the following Lyapunov function candidate

$$V_1 = \gamma_\alpha V_\alpha + \gamma_\beta V_\beta + \gamma_\gamma V_\gamma + V_{gn}, \tag{16}$$

where $\gamma_\alpha, \gamma_\beta, \gamma_\gamma > 0$ are weighting factors.

Differentiating (16) and substituting (9), (10), (13), and (14) into it, get

$$\dot{V}_1 = \gamma_\alpha \sum_{i=1}^{n}\sum_{j\neq i}^{n} \frac{\partial f_p(\| q_j - q_i \|, a_\alpha, b_\alpha)}{\partial(\| q_j - q_i \|)} \left[\frac{x_j - x_i}{\| q_j - q_i \|}(\dot{x}_j - \dot{x}_i) + \frac{y_j - y_i}{\| q_j - q_i \|}(\dot{y}_j - \dot{y}_i) \right]$$

$$+ \gamma_\beta \sum_{i=1}^{n}\sum_{k=1}^{m} \frac{\partial f_p(\| q_{v,k} - q_i \|, a_\beta, b_\beta)}{\partial(\| q_{v,k} - q_i \|)} \left[\frac{x_{v,k} - x_i}{\| q_{v,k} - q_i \|}(\dot{x}_{v,k} - \dot{x}_i) + \frac{y_{v,k} - y_i}{\| q_{v,k} - q_i \|}(\dot{y}_{v,k} - \dot{y}_i) \right]$$

$$+ \gamma_\gamma \sum_{i=1}^{n}\sum_{p\in\Omega(q_i)} \frac{\partial f_p(\| q_{i,p} - q_i \|, a_\gamma, b_\gamma)}{\partial(\| q_{i,p} - q_i \|)} \left[\frac{x_{i,p} - x_i}{\| q_{i,p} - q_i \|}(\dot{x}_{i,p} - \dot{x}_i) + \frac{y_{i,p} - y_i}{\| q_{i,p} - q_i \|}(\dot{y}_{i,p} - \dot{y}_i) \right]$$

$$+ \sum_{i=1}^{n} \left[\gamma_u (u_{lv} - u_{li})(\dot{u}_{lv} - \dot{u}_{li}) + \gamma_\psi (\psi_{lv} - \psi_{li})(\dot{\psi}_{lv} - \dot{\psi}_{li}) \right]. \tag{17}$$

For expression convenience, we apply the following simplifications in the remainder of this chapter that $\partial f_p(\zeta, a, b) = \partial f_p(\zeta, a, b)/\partial\zeta$, $\xi_{ji} = \xi_j - \xi_i$, $\xi_{vki} = \xi_{v,k} - \xi_i$, and $\xi_{ipi} = \xi_{i,p} - \xi_i$ with $\xi = q, x, y$, and $u_{lvi} = u_{lv} - u_{li}$, $\psi_{lvi} = \psi_{lv} - \psi_{li}$. As aforementioned, all virtual leaders have the same velocity and heading with the kinematics of $\dot{x}_{v,k} = u_{lv}\cos\psi_{lv}$ and $\dot{y}_{v,k} = u_{lv}\sin\psi_{lv}$. Substituting (5) into (17) and further expanding it, we can get

$$\dot{V}_1 = \gamma_\alpha \sum_{i=1}^{n}\sum_{j\neq i}^{n} \partial f_p(\| q_{ji} \|, a_\alpha, b_\alpha) \left[\frac{x_{ji}}{\| q_{ji} \|} \left[(\dot{x}_j - u_{lv}\cos\psi_{lv}) + (u_{lv}\cos\psi_{lv} - \dot{x}_i) \right] \right.$$

$$\left. + \frac{y_{ji}}{\| q_{ji} \|} \left[(\dot{y}_j - u_{lv}\sin\psi_{lv}) + (u_{lv}\sin\psi_{lv} - \dot{y}_i) \right] \right]$$

$$+ \gamma_\beta \sum_{i=1}^{n}\sum_{k=1}^{m} \partial f_p(\| q_{vki} \|, a_\beta, b_\beta) \left[\frac{x_{vki}}{\| q_{vki} \|}(u_{lv}\cos\psi_{lv} - \dot{x}_i) + \frac{y_{vki}}{\| q_{vki} \|}(u_{lv}\sin\psi_{lv} - \dot{y}_i) \right]$$

$$-\gamma_\gamma \sum_{i=1}^{n} \sum_{p\in\Omega(q_i)} \partial f_p(\| q_{ipi} \|, a_\gamma, b_\gamma)\left(\frac{x_{ipi}}{\| q_{ipi} \|}\dot{x}_i + \frac{y_{ipi}}{\| q_{ipi} \|}\dot{y}_i \right)$$

$$+\sum_{i=1}^{n}\left[\gamma_u u_{lvi}(\dot{u}_{lv}-\dot{u}_{li})+\gamma_\psi \psi_{lvi}(\dot{\psi}_{lv}-\dot{\psi}_{li})\right]$$

$$=2\gamma_\alpha \sum_{i=1}^{n}\sum_{j\neq i}^{n} \partial f_p(\| q_{ji} \|, a_\alpha, b_\alpha)\left[\frac{x_{ji}}{\| q_{ji} \|}(u_{lv}\cos\psi_{lv}-\dot{x}_i)+\frac{y_{ji}}{\| q_{ji} \|}(u_{lv}\sin\psi_{lv}-\dot{y}_i) \right]$$

$$+\gamma_\beta \sum_{i=1}^{n}\sum_{k=1}^{m} \partial f_p(\| q_{vki} \|, a_\beta, b_\beta)\left[\frac{x_{vki}}{\| q_{vki} \|}(u_{lv}\cos\psi_{lv}-\dot{x}_i)+\frac{y_{vki}}{\| q_{vki} \|}(u_{lv}\sin\psi_{lv}-\dot{y}_i) \right]$$

$$+\gamma_\gamma \sum_{i=1}^{n} \sum_{p\in\Omega(q_i)} \partial f_p(\| q_{ipi} \|, a_\gamma, b_\gamma)\left[\frac{x_{ipi}}{\| q_{ipi} \|}(u_{lv}\cos\psi_{lv}-\dot{x}_i)+\frac{y_{ipi}}{\| q_{ipi} \|}(u_{lv}\sin\psi_{lv}-\dot{y}_i) \right]$$

$$-\gamma_\gamma \sum_{i=1}^{n} \sum_{p\in\Omega(q_i)} \partial f_p(\| q_{ipi} \|, a_\gamma, b_\gamma)\left(\frac{x_{ipi}}{\| q_{ipi} \|}u_{lv}\cos\psi_{lv} + \frac{y_{ipi}}{\| q_{ipi} \|}u_{lv}\sin\psi_{lv} \right)$$

$$+\sum_{i=1}^{n}\left[\gamma_u u_{lvi}(\dot{u}_{lv}-\dot{u}_{li})+\gamma_\psi \psi_{lvi}(\dot{\psi}_{lv}-\dot{\psi}_{li})\right]$$

$$=\sum_{i=1}^{n}\Big[\Lambda_{xi}(u_{lv}\cos\psi_{lv}-u_{li}\cos\psi_{li})+\Lambda_{yi}(u_{lv}\sin\psi_{lv}-u_{li}\sin\psi_{li})+\Lambda_o$$

$$+\gamma_u u_{lvi}(\dot{u}_{lv}-\dot{u}_{li})+\gamma_\psi \psi_{lvi}(\dot{\psi}_{lv}-\dot{\psi}_{li})\Big]$$

$$=\sum_{i=1}^{n}\Bigg[(\Lambda_{xi}\cos\psi_{li}+\Lambda_{yi}\sin\psi_{li})u_{lvi}+2u_{lv}\left(\Lambda_{yi}\cos\frac{\psi_{lv}+\psi_{li}}{2}-\Lambda_{xi}\sin\frac{\psi_{lv}+\psi_{li}}{2}\right)\sin\frac{\psi_{lvi}}{2}$$

$$+\Lambda_o+\gamma_u u_{lvi}(\dot{u}_{lv}-f_{ui}\cos\psi_{ai}-f_{vi}\sin\psi_{ai}-b_{ui}\cos\psi_{ai}\cdot\tau_{ui})+\gamma_\psi \psi_{lvi}(\dot{\psi}_{lv}-\dot{\psi}_{ai}-\alpha_{ri}+e_{ri})\Bigg], \quad (18)$$

where $e_{ri}=\alpha_{ri}-r_i$ with α_{ri} a stabilizing function (Krstic et al., 1995) for virtual input r_i, Λ_{xi} and Λ_o are defined as following

$$\Lambda_{xi}=2\gamma_\alpha \sum_{j\neq i}^{n}\partial f_p(\| q_{ji} \|,a_\alpha,b_\alpha)\frac{x_{ji}}{\| q_{ji} \|}+\gamma_\beta \sum_{k=1}^{m}\partial f_p(\| q_{vki} \|,a_\beta,b_\beta)\frac{x_{vki}}{\| q_{vki} \|}$$

$$+\gamma_\gamma \sum_{p\in\Omega(qi)}\partial f_p(\| q_{ipi} \|,a_\gamma,b_\gamma)\frac{x_{ipi}}{\| q_{ipi} \|}, \quad (19)$$

$$\Lambda_o=-\gamma_\gamma \sum_{p\in\Omega(q_i)}\partial f_p(\| q_{ipi} \|, a_\gamma, b_\gamma)\left(\frac{x_{ipi}}{\| q_{ipi} \|}u_{lv}\cos\psi_{lv}+\frac{y_{ipi}}{\| q_{ipi} \|}u_{lv}\sin\psi_{lv} \right), \quad (20)$$

and Λ_{yi} is defined to have the same form with Λ_{xi} with only x displaced by y.

According to (18), in this step we choose the control laws as following

$$\tau_{ui} = b_{ui}^{-1} \sec\psi_{ai} \left[\dot{u}_{lv} - f_{ui} \cos\psi_{ai} - f_{vi} \sin\psi_{ai} + \gamma_u^{-1} \left(k_{ui} u_{lvi} + \Lambda_{xi} \cos\psi_{li} + \Lambda_{yi} \sin\psi_{li} \right) \right], \tag{21}$$

$$\alpha_{ri} = \dot{\psi}_{lv} - \dot{\psi}_{ai} + \gamma_\psi^{-1} \left[k_{\psi i} \psi_{lvi} + u_{lv} \left(\Lambda_{yi} \cos\frac{\psi_{lv} + \psi_{li}}{2} - \Lambda_{xi} \sin\frac{\psi_{lv} + \psi_{li}}{2} \right) \frac{\sin(\psi_{lvi}/2)}{\psi_{lvi}/2} \right], \tag{22}$$

where $k_{ui}, k_{\psi i} > 0$ are design parameters.

Remark 5. According to (2), we can see that $\dot{\psi}_{ai}$ contains the acceleration terms $\dot{u}$ and $\dot{v}$. However, since $f_{ui}(\cdot), f_{vi}(\cdot), f_{ri}(\cdot) \in C^2$, it is not difficult to verify that the surge force control law (21) is differentiable, and this means that $\dot{u}$ and $\dot{v}$ are also differentiable. Consequently, we can conclude that $\dot{\psi}_{ai}$ is also differentiable and so is α_{ri}.

Substituting (21) and (22) into (18), get

$$\dot{V}_1 = \sum_{i=1}^{n} \left(-k_{ui} u_{lvi}^2 - k_{\psi i} \psi_{lvi}^2 + \gamma_\psi e_{ri} \psi_{lvi} + \Lambda_o \right). \tag{23}$$

Step 2. Rewrite the final equation of (5) as following

$$\dot{e}_{ri} = \dot{\alpha}_{ri} - f_{ri} - b_{ri} \tau_{ri}. \tag{24}$$

Now, consider the following Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2} \sum_{i=1}^{n} e_{ri}^2. \tag{25}$$

Differentiating (25) and substituting (23)~(24) into it, finally we can have

$$\dot{V}_2 = \sum_{i=1}^{n} \left[-k_{ui} u_{lvi}^2 - k_{\psi i} \psi_{lvi}^2 + \gamma_\psi e_{ri} \psi_{lvi} + e_{ri} (\dot{\alpha}_{ri} - f_{ri} - b_{ri} \tau_{ri}) + \Lambda_o \right]. \tag{26}$$

According to (26), the control law for τ_{ri} is chosen as

$$\tau_{ri} = b_{ri}^{-1} \left(k_{ri} e_{ri} + \dot{\alpha}_{ri} - f_{ri} + \gamma_\psi \psi_{lvi} \right), \tag{27}$$

where $k_{ri} > 0$ is a design parameter. Substituting (27) into (26), finally we can have

$$\dot{V}_2 = \sum_{i=1}^{n} \left(-k_{ui} u_{lvi}^2 - k_{\psi i} \psi_{lvi}^2 - k_{ri} e_{ri}^2 + \Lambda_o \right). \tag{28}$$

According to *Assumption 3*, after a certain period of time such that $\min_{k,i} \| q_{v,k} - q_{i,p} \| \geq b_\beta + b_\gamma$, if all vehicles are still following the virtual leaders (this means $\| q_{kvi} \| \leq b_\beta$), then $\dot{f}_p(\| q_{ipi} \|, a_\gamma, b_\gamma) = 0$, $\forall i = 1, \cdots, n$ and $\forall p \in \Omega(q_i)$. Otherwise, we have $u_{lv} = 0$. In both cases, it is easy to verify that $\Lambda_o = 0$. Consequently, after a certain period of time, we always have $\dot{V}_2 \leq 0$, and $\dot{V}_2 = 0$ if and only if $u_{lvi} = \psi_{lvi} = e_{ri} = 0$, $\forall i = 1, \cdots, n$.

Theorem. Consider the schooling for multiple underactuated underwater vehicles whose kinematics and dynamics can be expressed as (1) with *Assumption 1~3*. If we choose the formation control laws as (21) and (27), then the schooling asymptotically converges to a

certain local minimum and all vehicles included in the group asymptotically move with the same velocity and heading while keeping obstacle avoidance.

Remark 6. Consider the case where one or more vehicles are trapped by obstacles. If we design V_β to take finite cut-off ($b_\beta < +\infty$), which is different from Olfati-Saber (2006) and Do (2007), then after $\| q_{vkj} \| \geq b_\beta$ with q_j trapped vehicles, we have $\dot{f}_p(\| q_{vkj} \|, a_\beta, b_\beta) = 0$. This means that after a certain period of time, only two kinds of potentials V_α and V_γ are remained to restrict the behaviour of the trapped vehicles. Therefore, we can guarantee the trapped vehicles to collision-free with obstacles even without *C7* in *Assumption 3*.

Remark 7. So far, we derive the schooling algorithms under the condition $u_i \geq u_{\min} \geq \rho v_{\max}$ as in *Assumption 1*. In fact, substituting (21) into (3), we have

$$\dot{u}_{lvi} = -\gamma_u^{-1} k_{ui} u_{lvi} - \gamma_u^{-1} \left(\Lambda_{xi} \cos\psi_{li} + \Lambda_{yi} \sin\psi_{li} \right). \tag{29}$$

On the other hand, if we take the smooth potential function as the form of (6), then it is not difficult to verify that ∂f_p is bounded. Further according to (19), we can see that Λ_{xi} and Λ_{yi} are also bounded. Therefore, through proper selection of weighting factors γ_α, γ_β and γ_γ, we can make $\Lambda_{xi} \cos\psi_{li} + \Lambda_{yi} \sin\psi_{li}$ arbitrarily small. Consequently, further through proper selection of k_{ui}, it is easy to get

$$| u_{lvi}(t) | \leq | u_{lvi}(0) |. \tag{30}$$

Therefore, if we design the virtual leaders' motions such that $u_{lv} \geq \sqrt{v_{\max}^2 + u_{\min}^2} + U$ with $U > 0$ a design parameter, then it is not difficult to verify that $\forall t \geq 0$, have $u_i(t) \geq u_{\min}$ under the initial condition $| u_{lvi}(0) | \leq U$. In other word, assumption of $u_i(t) \geq u_{\min}$ always can be guaranteed in practice under suitable selection of initial conditions. Meanwhile, this means that the proposed formation control method only guarantees local stability of the schooling. Here it is notable that $| u_{lvi}(0) | \leq U$ is a certain sufficient condition for $u(t) \geq u_{\min}$, not a necessary one.

Remark 8. Since polar coordinate transformation does not satisfy to be a diffeomorphism, usually $u_{lvi}, \psi_{lvi} \to 0$ does not guarantee the same convergence properties of $u_i \to u_v$, $v_i \to v_v$ and $r_i \to r_v$, where u_v, v_v, r_v are the virtual leader's velocity components. This may be one of disadvantages to applying polar coordinate transformation. However, in practical applications, it is much more difficult to design the motion of virtual leaders in the Cartesian frame. For example, for given u_v and r_v, because of high nonlinearity, it is difficult to directly calculate v_v through $\dot{v}_v = f_v(u_v, v_v, r_v)$. On the contrary, in the corresponding polar frame, we can easily bring out the reference paths for virtual leaders from given u_{lv} and ψ_{lv}.

5. Simulation studies

In this section, we carry out some simulation studies to illustrate the effectiveness of proposed vehicles schooling scheme. In the simulation, the vehicles group consists of three vehicles, and each of them is modelled as the six DOF nonlinear dynamics of ISiMI AUV (Lee et al., 2003), which has the similar mechanical structure as REMUS AUV (Prestero, 2001). For this ISiMI AUV, we use the saturation conditions as $| \tau_{ui} | \leq 50N$ and $| \delta_{ri} | \leq \pi / 6 rad$ in the simulation. And all virtual leaders are assumed move with the same velocity and heading as $u_{lv} = 1.54 m/s, \psi_{lv} = 0 rad$.

5.1 Straight line schooling

In this case, we can choose the three virtual leaders to locate at the vertices of a certain isosceles triangle as in Fig. 3 (a) with the initial positions as $q_{v1}(0)=(6,10)$, $q_{v2}(0)=(6,26)$ and $q_{v3}(0)=(34,18)$. Three vehicles' initial conditions are as: $q_1(0)=(0,24)$, $q_2(0)=(12,12)$, $q_3(0)=(28,16)$, $\psi_1(0)=0.1rad$, $\psi_2(0)=0.05rad$, $\psi_3(0)=0.15rad$, and $u_1(0)=u_2(0)=u_3(0)=0.5$ m/s with all other variables taking zero values. Other design parameters are taken as: $k_{ui}=8$, $k_{\psi i}=3$, $k_{ri}=12$, $i=1,\cdots,3$ and $\gamma_\alpha=0.3$, $\gamma_\beta=0.5$, $\gamma_u=100$, $\gamma_\psi=0.2$.

For potential functions V_α and V_β, both of them take the form as (6) and corresponding parameters are chosen as: $a_\alpha=12$, $b_\alpha=20$, $a_\beta=10$, $b_\beta=20$, $c_\alpha=2$, $c_\beta=3$ and $h=0.9$.

Simulation results are shown in Fig. 5 and 6. Fig. 5 shows the vehicles group schooling in its straight line movement. From Fig. 6, we can see that there is not any collision between vehicles in the schooling.

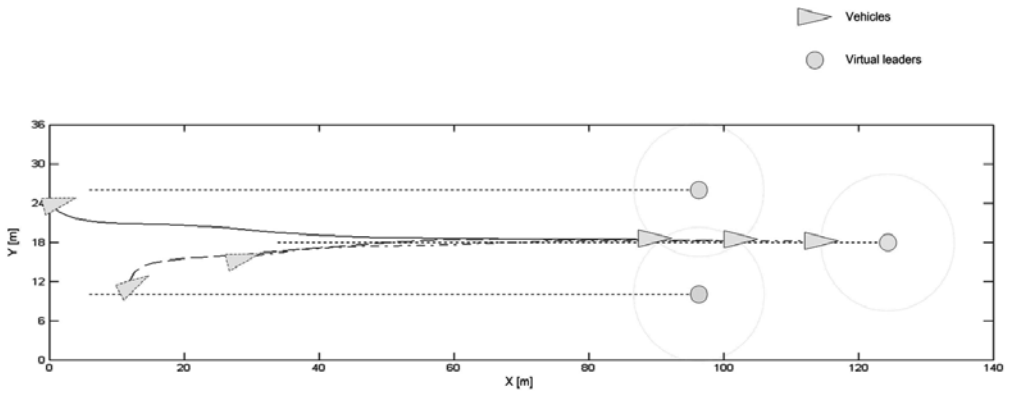


Fig. 5. Schooling of the vehicles in a straight-line movement.

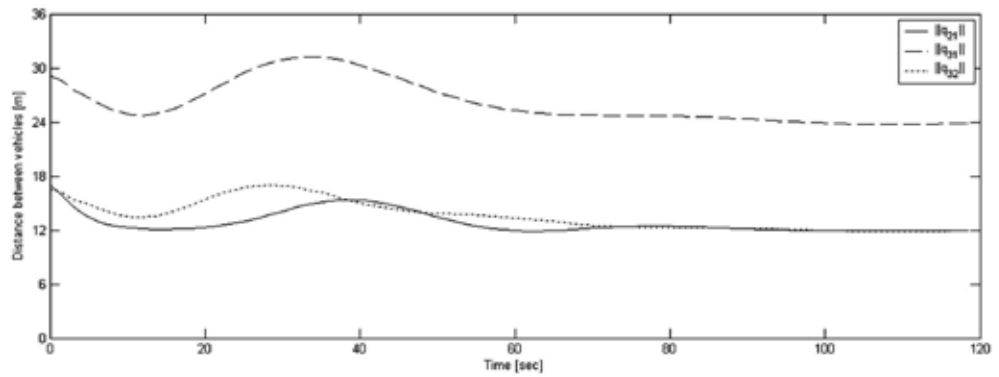


Fig. 6. Schooling geometry for a straight-line movement.

5.2 Triangular schooling

For triangular schooling, we locate the three virtual leaders as in Fig. 3 (b) with the initial positions taken as $q_{v1}(0)=(0,40)$, $q_{v2}(0)=(0,8)$ and $q_{v3}(0)=(20,24)$. Three vehicles' initial conditions are chosen as $q_1(0)=(9,39)$, $q_2(0)=(3,10)$, $q_3(0)=(10,22)$, $\psi_1(0)=0.1rad$, $\psi_2(0)=$

$0.05rad, \psi_3(0) = 0.15rad$, and $u_1(0) = u_2(0) = u_3(0) = 0.5 \quad m/s$ with all other variables taking zero values. Other design parameters are taken as $k_{ui} = 12, k_{\psi i} = 8, k_{ri} = 12, i = 1, \cdots, 3$ and $\gamma_\alpha = 0.3, \gamma_\beta = 0.5, \gamma_u = 60, \gamma_\psi = 0.3$.

Same as in the straight line case, both of V_α and V_β take the form as (6) with the parameters chosen as $a_\alpha = 12, b_\alpha = 15, a_\beta = 8, b_\beta = 20, c_\alpha = 2, c_\beta = 3$ and $h = 0.9$.

Corresponding simulation results are presented in Fig. 7 and 8. The vehicles schooling in the triangular movement is shown in Fig. 7 with no collision between any of two vehicles (see Fig. 8).

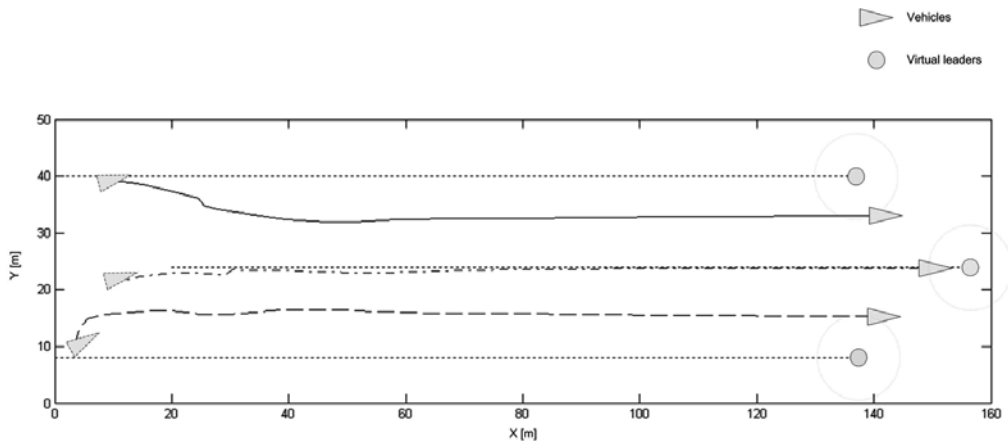


Fig. 7. Schooling of the vehicles in a triangular movement.

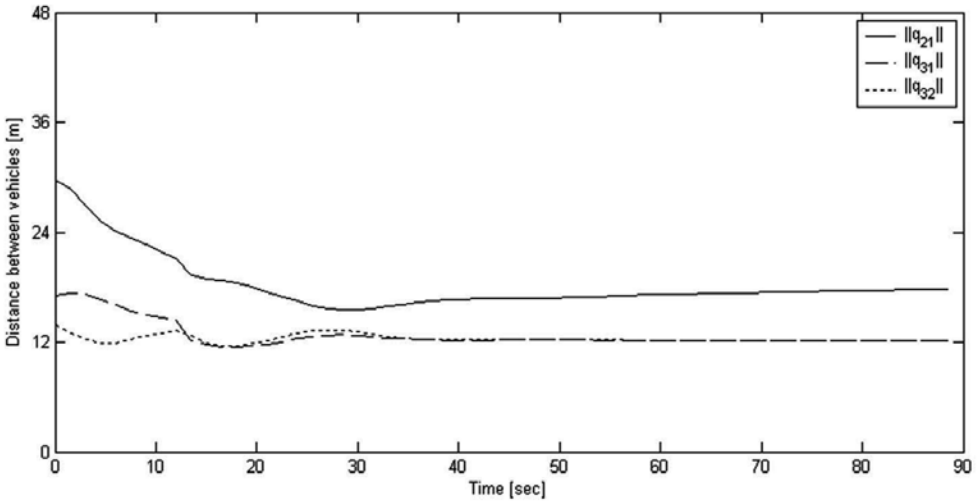


Fig. 8. Schooling geometry for a triangular movement.

5.3 Equilateral triangular schooling with obstacle avoidance

In this case, we consider an equilateral triangular schooling of the vehicles with obstacle avoidance. The obstacle is modelled as a circle located at $q_p = (40, 10)$ with radius as 3m. For

the schooling, two virtual leaders are chosen as in Fig. 3 (c) with the initial positions taken as $q_{v1}(0) = (2\sqrt{3}, 10)$ and $q_{v2}(0) = (10\sqrt{3}, 10)$. The vehicles' initial conditions are taekn as $q_1(0) = (0, 1)$, $q_2(0) = (2, 15)$, $q_3(0) = (7\sqrt{3}, 11)$, $\psi_1(0) = 0.1rad$, $\psi_2(0) = 0.05rad$, $\psi_3(0) = 0.15rad$ and $u_1(0) = u_2(0) = u_3(0) = 0.5m/s$ with all other variables taking zero values. Other design parameters are taken as $k_{ui} = 18$, $k_{\psi i} = 10$, $k_{ri} = 12$, $i = 1, \cdots, 3$ and $\gamma_\alpha = 0.15$, $\gamma_\beta = 0.2$, $\gamma_\gamma = 0.15$, $\gamma_u = 240$, $\gamma_\psi = 6$.

Both V_α and V_β are also taken as the form as (6) with the parameters as $a_\alpha = 12$, $b_\alpha = 30$, $a_\beta = 4\sqrt{3}$, $b_\beta = 30$, $a_\gamma = b_\gamma = 6$, $c_\alpha = 4$, $c_\beta = 5$, $c_\gamma = 90$ and $h = 0.9$.

Simulation results are depicted in Fig. 9~12. Fig. 9 shows the vehicles schooling in the equilateral triangular movement with obstacle avoidance. From Fig. 10, we can see that there is not any collision between vehicles. Fig. 11 presents the vehicles' velocity and heading matching in the schooling, and Fig. 12 shows the histories of proposed formation control laws for τ_{ui} and δ_{ri}.

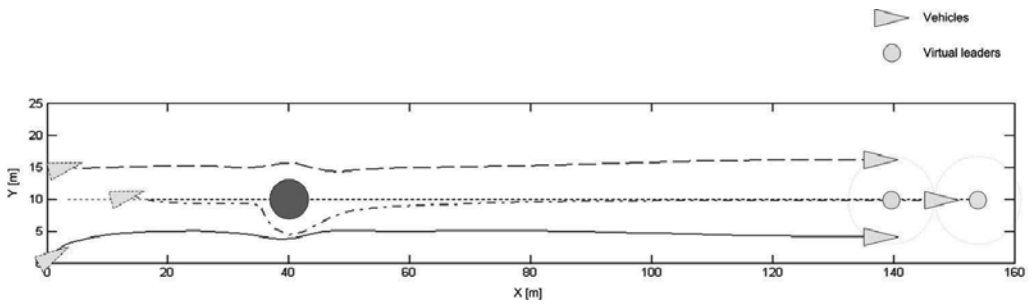


Fig. 9. Schooling of the vehicles in an equilateral triangular movement with obstacle avoidance.

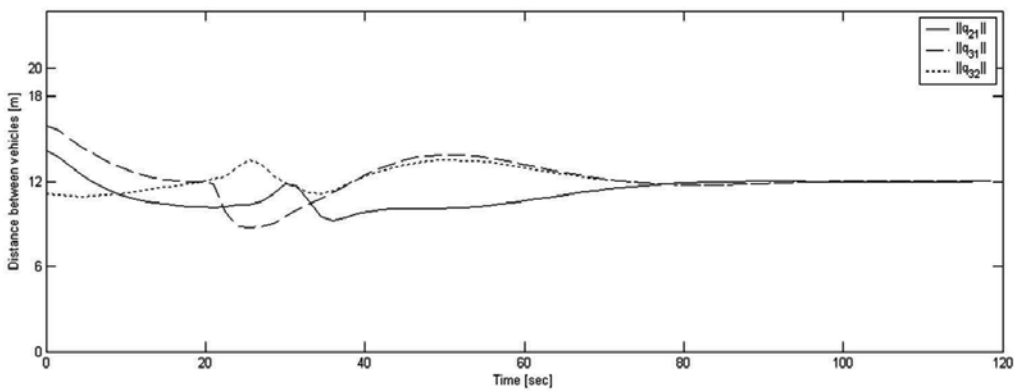


Fig. 10. Schooling geometry for an equilateral triangular movement with obstacle avoidance.

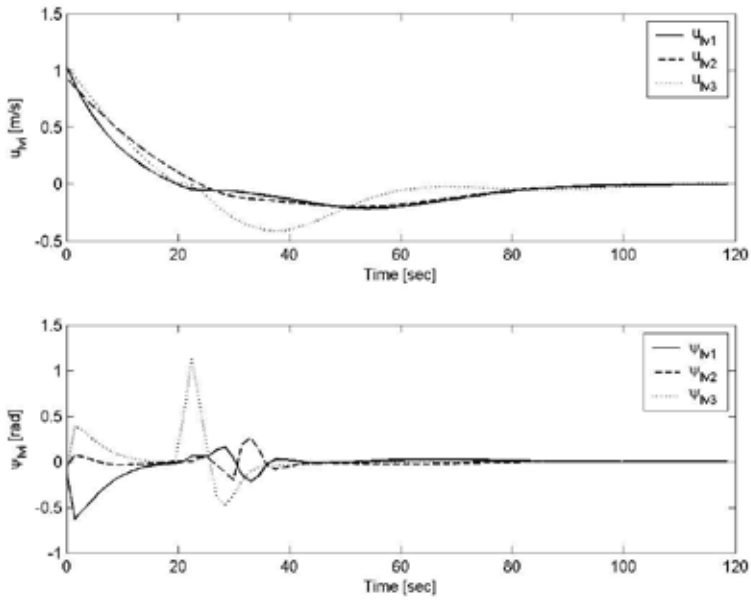


Fig. 11. Group velocity and heading matching.

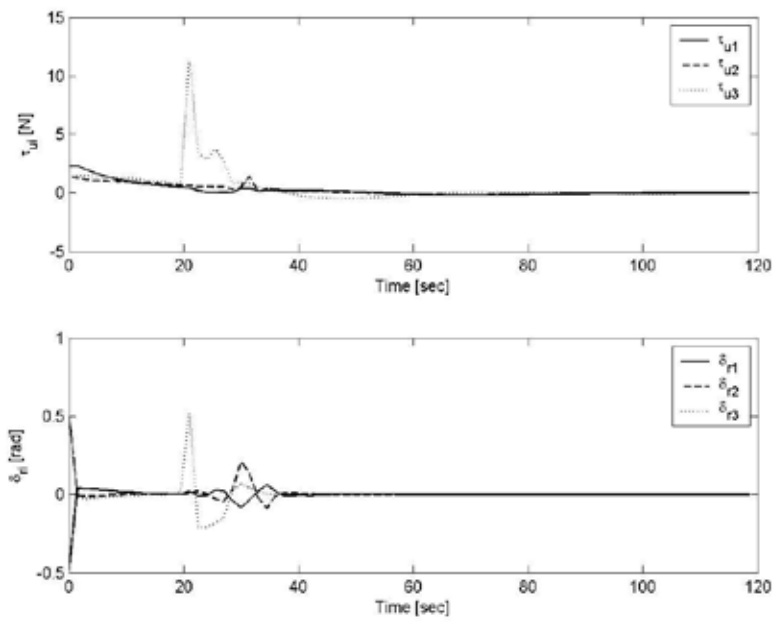


Fig. 12. Histories of proposed formation control laws.

6. Summary

In this chapter, we have investigated an asymptotic schooling scheme for multiple underactuated underwater vehicles. For each vehicle, there are only two control inputs – surge force and yaw moment available for its three DOF motion in the horizontal plane. The main difficulty in the tracking of this kind of vehicle is how to properly handle the vehicle's sway dynamics. To deal with this problem, in this chapter, we have introduced a certain polar coordinates transformation, through which the vehicle's dynamics can be reduced to a two-inputs strict-feedback form. The vehicles schooling has been conducted by properly selected smooth potential function, which consists of three different parts: one is for the interaction between vehicles, another is for group navigation, and the third one is for obstacle avoidance. The proposed formation algorithm guarantees the vehicles asymptotic schooling and velocity and heading matching while keeping obstacle avoidance.

Proposed schooling scheme has been derived under the condition of $u(t) \geq u_{\min} > 0$, which inversely can be guaranteed by proposed formation control laws being combined with some suitable initial conditions. Therefore, the proposed schooling method only can guarantee the local stability. Moreover, it is notable that the following issues should be considered in our future works.

- Finite cut-off ($b < +\infty$ in *Definition 1*) of potential function, which was applied in the previous works (Leonard and Fiorelli, 2001; Olfati-Saber, 2006; Do, 2007), also plays an important role in the vehicles schooling in this chapter. However, since $b < +\infty$, it is easy to verify that $\partial f_p(\zeta, a, b)/\partial\zeta = 0$ if $\zeta \geq b$. For this reason, the proposed schooling scheme only guarantees certain local minimum. It is of interest to upgrade the present result to the one where the global minimum can be guaranteed in our future works.
- Another practical concern is for the robustness of proposed schooling scheme. In practice, there various uncertainty terms have to be faced, such as vehicle's modelling error, measurement noise, and disturbance, etc. All of these terms should be considered in our future practical applications.

7. Acknowledgements

This work was supported by the Ministry of Land, Transport and Maritime Affairs in Korea under Grant PMS162A and by the Korea Ocean Research & Development Institute under Grant PES120B.

8. References

Bacciotti. A. & Rosier, L. (2005). *Liapunov Functions and Stability in Control Theory*. Springer-Verlag, Berlin, Heidelberg, 2005

Brockett, R. W. ; Mullman, R. S. & Sussmann, H. J. (1983). *Differential Geometric Control Theory*. Boston, MA : Birkhauser, 1983

Do, K. D. (2007). Bounded controller for formation stabilization of mobile agents with limited sensing ranges. *IEEE Transactions on Automatic Control*, Vol. 52, No. 2, pp. 569-576

Do, K. D. & Pan, J. (2005). Global tracking control of underactuated ships with nonzero off-diagonal terms in their system matrices. *Automatica*, Vol. 41, No. 1, pp. 87-95

Do, K. D. ; Jiang, Z. P. & Pan, J. (2002a). Underactuated ship global tracking under relaxed conditions. *IEEE Transactions on Automatic Control*, Vol. 47, No. 9, pp. 1529-1536

Do, K. D. ; Jiang, Z. P. & Pan, J. (2002b). Universal controllers for stabilization and tracking of underactuated ships. *Systems & Control Letters*, Vol. 47, No. 4, pp. 299-317

Do, K. D. ; Jiang, Z. P. & Pan, J. (2004). Robust adaptive path following of underactuated ships. *Automatica*, Vol. 40, Nol 6, pp. 929-944

Dunbar, W. B. & Murray, R. M. (2002). Model predictive control of coordinated multi-vehicle formations, *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 4631-4636, Las Vegas, Nevada, USA, December 2002

Edwards, D. D. ; Bean, T. A. ; Odell, D. L. & Anderson, M. J. (2004). A leader-follower algorithm for multiple AUV formations, *Proceedings of Workshop on Autonomous Underwater Vehicles, 2004 IEEE/OES*, pp. 40-46, Sebasco Estates, Maine, USA, June 2004

Fax, J. A. & Murray, R. M. (2004). Information flow and cooperative control of vehicle formations. IEEE Transaction on Automatic Control, Vol. 49, No. 9, pp. 1465-1476,

Fiorelli, E. ; Leonard, N. E. ; Bhatta, P. ; Paley, A. ; Bachmayer, R. & Fratantoni, D. M. (2006). Multi-AUV control and adaptive sampling in Monterey Bay. *IEEE Journal of Oceanic Engineering*, Vol. 31, No., 4, pp. 935-948

Fossen, T. I. (2002). *Marine Control Systems*. Trondheim, Norway : Marine Cybermetics, 2002

Fredriksen, E. & Pettersen, K. Y. (2006). Global k-exponential way-point maneuvering of ships : Theory and experiments. *Automatica*, Vol. 42, No. 4, pp. 677-687

Gue, J. ; Wei, Y. Y. ; Chiu, F. C. & Cheng, S. W. (2004). A maximum entropy method for multi-AUV grouping, *Proceedings of IEEE/MTS Oceans'04*, pp. 532-536, Kobe, Japan, November 2004

Jiang, Z. P. (2002). Global tracking control of underactuated ships by Lyapunov's direct method. *Automatica*, Vol. 38, No. 2, pp. 301-309

Jiang, Z. P. & Nijmeijer, H. (1999). A recursive technique for tracking control of nonholonomic systems in chained form. *IEEE Transactions on Automatic Control*, Vol. 44, No. 2, pp. 265-279

Krstic, M. ; Kanellakopoulos, I. & Kokotovic, P. (1995). *Nonlinear and Adaptive Control Design*. John Wiley & Sons, Inc., New York, 1995

Latombe, J. (1991). *Robot Motion Planning*. Norwell, MA :Kluwer, 1991

Lee, P. M. et al. (2003). *Development of an Advanced Deep-Sea Unmanned Underwater Vehicle (II)*. Technical Report, UCM0043A-2442, KORDI, Daejeon, Korea, 2003

Leonard, N. E. & Fiorelli, E. (2001). Virtual leaders, artificial potentials and coordinated control of groups, *Proceedings of the 40th IEEE Conference on Decision and Control*, pp. 2968-2973, Orlando, Florida, USA, December 2001

Li, J. H. & Lee, P. M. (2008). Stable schooling for multiple underactuated AUVs, *Proceedings of IFAC World Congress*, pp. 15022-15027, Seoul, Korea, July 2008

Marthiniussen, R. ; Vestgard, K. ; Klepaker, R. A. & Storkersen, N. (2004). HUGIN-AUV concept and operational experience to date, *Proceedings of IEEE/MTS Oceans'04*, pp. 846-850, Kobe, Japan, November 2004

Murray, R. M. & Sastry, S. S. (1993). Nonholonomic motion planning : Steering using sinusoids. *IEEE Transactions on Automatic Control*, Vol. 38, No. 5, pp. 700-716

Newman, J. N. (1977). Marine Hydrodynamics. The MIT Press, Cambridge, Massachusetts, USA, and London, England, 1977

Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems : algorithms and theory. *IEEE Transactions on Automatic Control*, Vol. 51, No. 3, pp. 401-420

Olfati-Saber, R. & Murray, R. M. (2002). Distributed cooperative control of multiple vehicle formations using structural potential functions, *Proceedings of 15th IFAC World Congress*, Barcelona, Spain, July 2002

Olfati-Saber, R. & Murray, R. M. (2003). Flocking with obstacle avoidance : cooperation with limited communication in mobile networks, *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 2022-2028, Maui, Hawaii, USA, December 2003

Pettersen, K. Y. & Nijmeijer, H. (2001). Underactuated ship tracking control : Theory and experiments. *International Journal of Control*, Vol. 74, No. 14, pp. 1435-1446

Prestero, T. (2001). *Verificatin of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicles*. Masters Thesis, Department of Ocean Engineering and Mechanical Engineering, MIT, 2001

Rey nolds, C. W. (1987). Flocks, herds, and schools : A distributed behavioral model, *Proceedings of ACM SIGGRAPH'87*, pp. 25-34, Anaheim, California, USA, July 1987

Reyhanoglu, M. (1997). Exponential stabilization of an underactuated autonomous surface vessel. *Automatica*, Vol. 33, No. 12, pp. 2249-2254

Rimon, E. & Koditschek, D. E. (1992). Exact robot navigation using artificial potential function. *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 5, pp. 501-518

Smith, T. R. ; Hanbmann, H. & Leonard, N. E. (2001). Orientation control of multiple underwater vehicles with symmetry-breaking potentials, *Proceedings of the 40th IEEE Conference on Decision and Control*, pp. 4598-4603, Orlando, Florida, USA, December 2001

Sousa, J. B. & Pereira, F. L. (2002). Coordinated control strategies for networked vehicles : an application to autonomous underwater vehicles, *Proceedings of International Workshop on Generalized Solutions in Control Problems*, Perslavl-Zalessky, Russia, July 2002

Watanabe, K. & Nakamura, A. (2005). A design of tiny basin test-bed for AUV multi agent, *Proceedings of IEEE/MTS Oceans 2005*, pp. 1002-1008, Washington D. C., USA, September 2005

Westwood, R.; Westwood, J. & Newman, P. (2007). *The AUV Gamechanger Report 2008-2017*. Douglas-Westwood Limited, Canterbury, UK, 2007

17

MARES – Navigation, Control and On-board Software

Aníbal Matos and Nuno Cruz
Universidade do Porto
Portugal

1. Introduction

Autonomous underwater vehicles (AUVs) are an emerging technology with applications in very different fields, such as military, homeland defence, underwater surveys, environment monitoring, and oceanographic studies (Griffiths, 2003). Although the use of AUVs in some of these fields is already well established, there is still a great research effort in areas related to the design and operation of these vehicles. The Ocean Systems Group at FEUP (Faculty of Engineering at the University of Porto) and ISR – Porto (Institute for Systems and Robotics – Porto) conducts research activities in marine robotics and has accumulated expertise in the utilization of AUVs and in the development of particular subsystems. This chapter addresses the design of the navigation and control systems of the MARES AUV (Cruz & Matos, 2008), a state-of-the-art small size AUV developed by the authors and already demonstrated at sea operations in 2007. The implementation of these systems in the vehicle on-board software is also discussed.

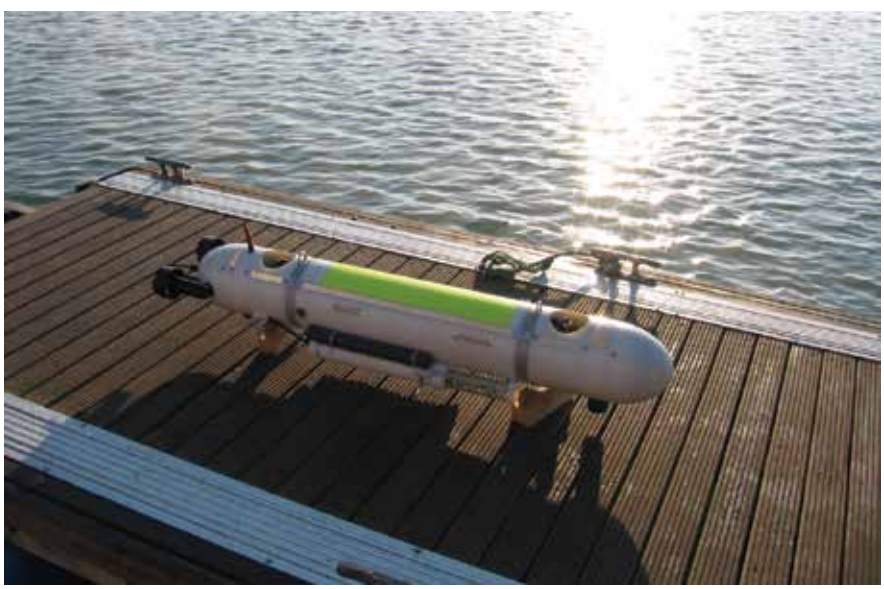


Fig. 1. MARES AUV ready for an open sea mission.

2. MARES AUV

MARES, or *Modular Autonomous Robot for Environment Sampling* (Fig. 1), is a 1.5m long AUV, designed and built by the Ocean Systems Group. The vehicle can be programmed to follow predefined trajectories, while collecting relevant data with the onboard sensors. MARES can dive up to 100m deep, and unlike similar-sized systems, has vertical thrusters to allow for purely vertical motion in the water column. Forward velocity can be independently defined, from 0 to 2 m/s. Major application areas include pollution monitoring, scientific data collection, sonar mapping, underwater video or mine countermeasures.

MARES configuration can change significantly according to the application scenario, so that it is difficult to define what is a *standard* configuration. In table 1 we summarize the main characteristics of the AUV version that was demonstrated at sea in November 2007.

Length	1.5 m
Diameter	20 cm
Weight in air	32 kg
Depth rating	100 m
Propulsion	2 horizontal + 2 vertical thrusters
Horizontal velocity	0-2 m/s, variable
Energy	Li-Ion batteries, 600 Wh
Autonomy/Range	about 10 hrs / 40 km

Table 1. MARES main characteristics.

2.1 Mechanical

All mechanical parts were designed using Solidworks® CAD software (Fig. 2) and machined from polyacetal in a local machine shop, with small parts in aluminium and stainless steel. Polyacetal is a high performance polymer, with a high degree of rigidity and mechanical strength that makes it an excellent weight-saving metal replacement. It is completely corrosion proof and it is readily available in a wide range of sizes of tubes and rods, at reasonable prices.

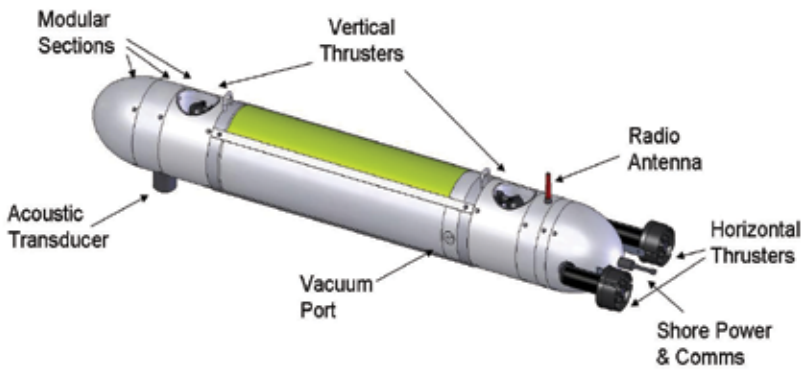


Fig. 2. MARES CAD model.

The vehicle hull evolves around a central watertight cylinder, where all electronic boards are installed, with the battery packs located at the bottom to lower the center of mass. To

simplify the design, this is the only watertight enclosure and therefore all other equipment has to be waterproof. The other polyacetal sections are designed to carry wet sensors and thrusters and they are fully interchangeable. This allows for very easy sensor swapping and/or repositioning, or even to test different configurations of thrusters. The main cylinder has 9 holes in each end cap, to accommodate standard bulkhead connectors and at the moment there are still several unused, sealed with dummy plugs.
The overall vehicle shape resembles that of a torpedo, with ellipsoids both at the nose cone and at the tail. This configuration is very simple to construct and allows for the vehicle length to be easily extended, as compared to other hull shapes without constant cross-sections. The central cylinder provides most of the vehicle flotation and it is also possible to increase its length, for example if more batteries are needed.
Typical small-size AUVs use vertical and horizontal fins to adjust heading and pitch, but this requires a minimum forward velocity for the control surfaces to be effective (von Alt et al., 1994; Crowell, 2006). On MARES, four independent COTS thrusters provide attitude control both in the horizontal and in the vertical plane. Two horizontal thrusters located at the tail control both forward velocity and rotation in the horizontal plane, while another set of thrusters, in the vertical direction, control vertical velocity and pitch angle. This arrangement permits operations in very confined areas, with virtually independent horizontal and vertical motion at velocities starting at 0 m/s. This is one of MARES innovations, as it cannot be seen in any AUV of similar size and weight. Furthermore, the modularity of the system allows the integration of other thrusters, for example to provide full control of the lateral motion.
It should be stressed that fins are usually more efficient for diving than thrusters, but with simple fins it is not possible to control pitch angle independently of depth. In mission scenarios where bottom tracking is important, such as sonar or video acquisition, a fin controlled AUV will pitch up and down to follow the terrain, affecting data quality. On the contrary, MARES AUV can control both pitch angle and depth independently, being able to maintain data quality even if the terrain has significant slopes.
Another advantage of using thrusters is that all moving parts can be fully shrouded and there are no fins protruding from the hull, which minimizes the risk of mechanical failure. In the end, we deliberately traded some of the efficiency with increased maneuverability and robustness.

2.2 Power and energy

Most of the power required by an AUV is spent in propulsion, with only a small amount permanently needed for onboard electronics. In MARES, all energy is stored in rechargeable Li-Ion battery packs, currently with a total amount of 600 Wh, at 14.4 V. Battery power is directly available to the motor controllers and, through a set of voltage converters, to the rest of the onboard electronics.
Battery endurance greatly depends on vehicle velocity, both in the horizontal and in the vertical plane. For typical *horizontal missions*, with relatively slow changes in depth, there is sufficient energy for about 8-10 hours of continuous operation (around 20-25 miles or 40 km). These are relatively modest numbers, but they seem to be sufficient for the great majority of envisaged missions. In any case, there is still some available volume for a few more battery packs. It should be stressed that these numbers refer to standard horizontal motion and it is also necessary to account for any significant vertical motion. For example,

the vehicle can hover almost motionless in the water column, at a specific depth, but still requiring some small amount of power to provide depth corrections. In this case, the total endurance will be longer in time but relative to a shorter horizontal range.

2.3 Computational system

The onboard computational system is based on a PC104 stack (Fig. 3), with a power supply board, a main processor board, and additional boards to interface with peripherals, such as health monitoring systems, actuation devices, and navigation and payload sensors. A flash disk is used to store both the onboard software and also the data collected during operations.

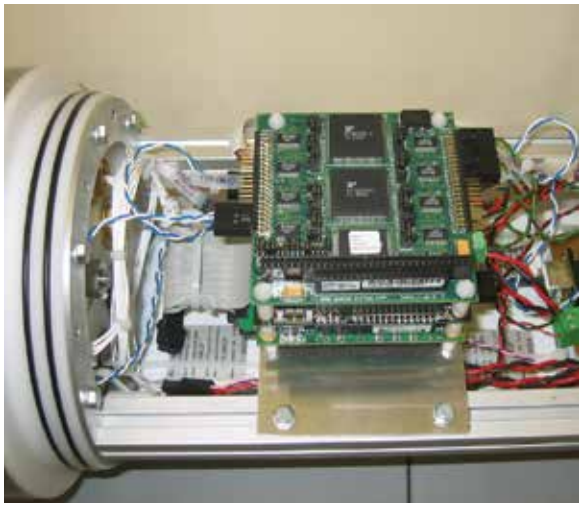


Fig. 3. MARES on-board computer.

2.4 Payload

The modularity of the vehicle allows for a simple integration of different payload sensors, involving three sub-tasks: mechanical installation, electronics interfacing and software.

Mechanically, a new sensor may be installed in a dedicated section of the hull, if it is relatively small. Alternatively, it can be externally attached to the vehicle body, since there are many fixing points available. In any case, it is important to verify the weight of the sensor (and adapter) in the water, to compensate with extra flotation if necessary. Naturally, the overall vehicle trim has also to be adjusted, particularly in the case of bulky or heavy payloads.

Most of the payload sensors transported by the AUV need energy and a communications link with the onboard computer. MARES has several spare connectors on both end caps of the main electronics compartment, that can be wired to provide power and receive data from these sensors. At the same time, the computational system has spare communication ports, easily configurable according to the payload specs.

As far as software is concerned, the integration of a new payload sensor requires the development of a dedicated software module, known as a *device driver*. Device drivers establish a communication link between the sensor and the onboard software core, allowing for the configuration of the sensor as well as data logging.

Naturally, these tasks are greatly reduced after the first time the sensor is tested. Since then, it becomes very simple to swap payload, just by integrating the proper set: sensor, electronics and software.

3. Navigation

The design of the navigation system of an AUV has to take into account several issues such as desired positioning and attitude accuracy, size, weight, and power consumption of available sensors and systems, and also overall cost. Available technologies include inertial navigation systems, digital compasses, tilt sensors, pressure cells, acoustic positioning systems, and Doppler based velocity meters. The MARES navigation package was selected taking into account the above mentioned issues but also the envisaged missions and application scenarios.

3.1 Sensors and systems

To estimate its position in real time, the vehicle carries a pressure sensor, a digital compass with a set of tilt sensors, and an acoustic system for long baseline (LBL) positioning (Vaganay et al., 1996). Vehicle depth is directly given by the pressure cell, while roll and pitch angles are obtained from the set of tilt sensors. The estimation of the horizontal position and velocity also employs dead-reckoning data.

The pressure cell has a centimetre level accuracy and outputs new data at a rate exceeding 100 Hz. This allows for a quite accurate vertical positioning and also for a software based estimation of heave velocity. The digital compass and tilt sensors unit provides data at 20 Hz with accuracies better than 0.5^{o}.

The acoustic system is a second generation of multi frequency boards, developed by the Ocean Systems Group, following some excellent results with previous versions (Cruz et al., 2001). These boards are installed on-board the vehicle but also on the acoustic beacons that are deployed in the operation area. This system is completely reconfigurable (pre mission or on line programming of detection/reply frequency pairs, channel sensitivity, etc.), allows complete control of signal transmission times, and provides access to low level signal detection data.

3.2 Horizontal position estimation

The real time estimation of the horizontal position of MARES is computed by a Kalman filter based algorithm (Matos et al., 1999) that combines dead-reckoning data (vehicle surge velocity and heading) with range measurements based on times of flight of acoustic signals. For a typical operation, two acoustic beacons are deployed in the operation area in a way such that the AUV, during the execution of its mission, does not cross the line connecting them. In this way, the range measurements between the AUV and both beacons unambiguously determine its horizontal position. Usually, these beacons are attached to surface buoys and signal detection and transmission time are transmitted to a shore station by a wireless link. This information allows for the real time external tracking of the AUV according to the algorithm proposed in (Cruz et al., 2001).

Since velocity data is obtained with respect to water, and no direct measurement of water currents is available, the horizontal components of such current are also estimated in real time. The navigation algorithm periodically updates the real time estimates of the horizontal

Fig. 4. Surface buoy attached to an acoustic beacon.

position of the vehicle (x, y) and of the water current (c_x, c_y) at a 20 Hz rate, according to the following dynamic system,

$$\begin{aligned} \dot{x} &= v\cos\psi + c_x \\ \dot{y} &= v\sin\psi + c_y \\ \dot{c}_x &= 0 \\ \dot{c}_y &= 0 \end{aligned} \tag{1}$$

At the same time, the associated error covariance matrix is updated. Whenever a new range measurement is obtained by the vehicle, the state variables and the error covariance matrix are corrected accordingly. This is accomplished by an iterative procedure based on the extended Kalman filter algorithm (Gelb, 1994), since range measurements are related to the state variables of the filter by the nonlinear relationship

$$r_i = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} \tag{2}$$

where (x, y, z) is the 3D position of the AUV, (x_i, y_i, z_i) is the 3D position of beacon i, and r_i is the predicted range measurement. Fig. 5 presents the range measurements between the AUV and two navigation beacons during an autonomous operation. It shows erroneous measurements caused multipath propagation of acoustic waves, as well as intervals of time when there are no measurements from at least one of the beacons, typically due to adverse propagation conditions. In order to increase the accuracy of the horizontal positioning, a validation mechanism is used to maximize the probability of rejecting those erroneous measurements.

Even in ideal propagation conditions, and neglecting error sources associated with the electronics of the transmission and reception acoustic boards, acoustic range measurements are directly affected by the sound speed in water. This, is turn, depends on the characteristics of the medium (temperature, salinity, pressure), and variations as high as 2%

are not unusual. It is therefore mandatory to determine the sound speed in the operation area and adjust range measurements accordingly. This can be done either by comparing range measurements close to the surface with DPGS based distance measurements, or by measuring the most relevant water characteristics with a CTD (conductivity, temperature, and depth) sensor. These calibration procedures allow horizontal position accuracies about 1 to 2 meters with respect to the positions of the beacons.

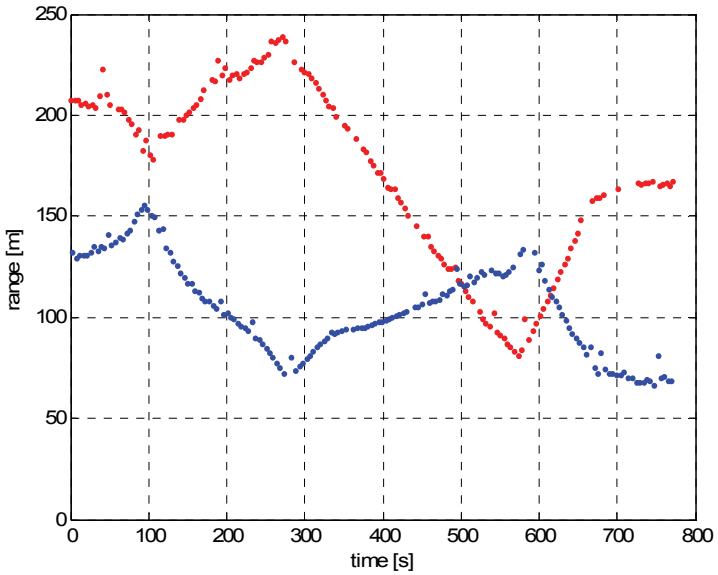


Fig. 5. Acoustic range measurements.

All MARES navigation data is stored in the flash disk carried by the vehicle, and real time GPS positioning data from the surface buoys attached to the acoustic beacons is transmitted to a shore station where it is also stored. This allows the post processing of all navigation data by a smoothing algorithm, therefore improving the accuracy of the vehicle positioning (Matos et al., 2003) and the space location of collected payload data.

4. Control

The general dynamic model of an underwater vehicle follows the 6 degrees of freedom template presented in (Fossen, 1994). It is a complete model that takes into account all the forces and moments acting on a submerged body, but for which it is not easy to design adequate controllers. Typically, AUVs evolve according to two dimensional motions, either in the vertical or in the horizontal plane. Therefore, the traditional approach for the control of theses vehicles is based on mode decoupling (Healey & Lienard, 1993).

Following such approach, the MARES control system is organized into four basic controllers: surge, heading, pitch and depth. The first two determine the horizontal motion of the vehicle and their outputs are combined to obtain the actuation for the horizontal thrusters. The outputs of the pitch and depth controllers are combined to provide the actuation for the vertical thrusters. Each one of the four basic controllers can operate in either closed or open loop mode.

4.1 Controllers

At the lowest level, all four controllers are assumed independent. Surge is controlled by the common mode of horizontal thrusters, heading is controlled by the differential mode of these thrusters, while depth is controlled by the common mode of vertical thrusters, and pitch by their differential mode. In this approach, the couplings between modes are treated as external disturbances, which must be taken into account when designing decoupled feedback controllers. This typically results in a small reduction of performance, which is largely balanced by the simplicity of the design and by the modularity of the approach.
Each one of these controllers can operate in different modes, ranging from a pure open loop operation to more complex structures with more than one feedback loop, as follows:

Surge:

- Open loop – input is a direct common mode command for the horizontal thrusters
- Velocity loop – input is a surge velocity reference

Heading:

- Open loop – input is a direct differential mode command for the horizontal thrusters
- Velocity loop – input is a heading rate reference
- Position loop – input is a heading reference
- Line tracking loop – input is a directed horizontal line reference

Depth:

- Open loop – input is a direct common mode command for the vertical thrusters
- Velocity loop – input is a heave velocity reference
- Position loop – input is a depth reference

Pitch:

- Open loop – input is a direct differential mode command for the vertical thrusters
- Velocity loop – input is a pitch rate reference
- Position loop – input is a pitch reference

4.2 Mission plan and elemental maneuvers

The autonomous operation of MARES is defined by a mission plan. Besides configuring a large set of variables that affect the vehicle behaviour (such as controller gains, maximum operating depth, operating frequencies of the acoustic systems, etc.), the mission plan also includes a set of elemental maneuvers that the vehicle should execute in sequence.
Each maneuver prescribes the behaviour all the four basic controllers (therefore defining the vehicle motion). It also defines its end condition, and a timeout for safety reasons. Besides some maneuvers that are mainly used for debugging purposes, the basic MARES maneuvers are:

- **dive** – a downwards maneuver, typically executed at the start of a mission or in depth transitions;
- **surface** – an upwards maneuver, typically executed at the end of a mission or in depth transitions;
- **hovering** – a maneuver that stops the vehicle at the current position;
- **gotoxy** – a horizontal plane maneuver that drives the vehicle along a straight line.

The possibility of independently defining each basic controller allows for very different vehicle behaviours, making the operation of MARES very flexible. For example, a pure vertical motion can be easily obtained by a dive maneuver with a closed loop pitch with zero reference, and a zero surge command; a combined vertical and horizontal motion can be achieved with a dive maneuver with a closed loop pitch with a negative (downward looking) reference and an appropriate surge command.
Furthermore, each basic controller is already prepared to accept inputs defined by external processes. This allows for the implementation of unconventional guidance strategies which can be based on payload data collected in real time.

5. On-board software

The onboard software was developed in C++, runs on a Linux kernel, and is composed by a set of independent processes. In this way, not only the system modularity and robustness are increased but also its debugging and recovery from unexpected events are much simpler. Communications between the modules rely on a message passing mechanism, using the User Datagram Protocol. This allows connectionless data transmissions, with reduced processing overhead, as required in this kind of applications. The data path between different processes can be easily reconfigured, increasing therefore the flexibility of the system and simplifying the integration of new modules. Moreover, it is also possible to modify data paths when the software is running by issuing *special* commands to the relevant modules.

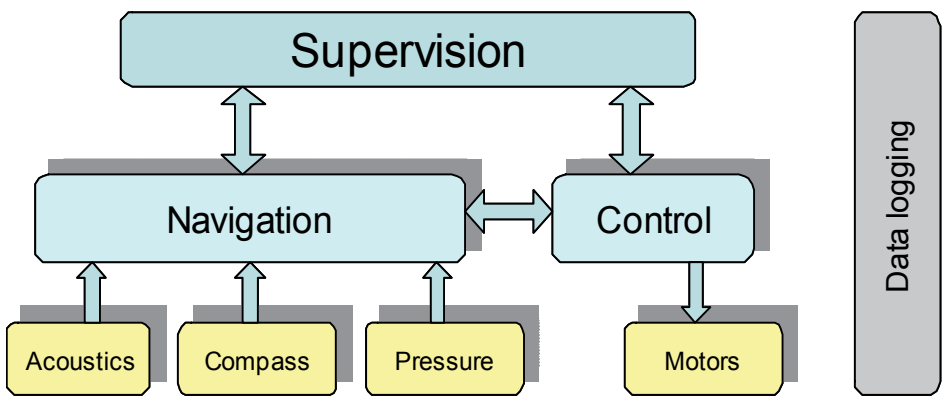


Fig. 6. Basic on-board software structure.

The basic on-board software follows the structured depicted in Fig. 6. The interface with the hardware is managed by dedicated processes that provide an abstraction layer. Processes that deal with navigation sensors and systems transmit their data to the navigation module. This one implements all necessary algorithms to estimate the AUV state in real time. It also sends such estimate to the control module.
The control module is responsible for the execution of the mission. In each control cycle it verifies the completion of the current maneuver and schedules the next one whenever the current ending conditions are met. This module also executes the procedures required by the different control loops.

A supervision module continuously monitors the behaviour of the vehicle and aborts the autonomous operation if safety margins are exceeded or unexpected events occur. This module can also be used to configure the whole software of the vehicle. For that purpose, it establishes a communication link with a shore control station whenever the vehicle is at the surface.
A black box data logging system registers all information related to the vehicle operation on the flash disk. The information includes raw data from the navigation sensors and systems, health monitoring data, processed navigation data and control data. This black box system is also prepared to register payload data.
Payload sensors and systems are typically controlled by additional dedicated modules. In general, these modules interact with the supervision module, for configuration and communication with the control station, and also with the data logger.
Additional modules implementing sensor based control strategies or other advanced features are easily integrated with the basic software structure. Besides programming and installing the new module, it is only necessary to redefine data paths, assuring that required sensor data is also sent to the new module, and that the new module sends its output data to the control module. The operation of such new module is then configured in the mission plan.

6. Conclusions

The first MARES water tests were conducted in a local pool in late 2006. Those tests served to validate the integrity of the system, adjust buoyancy and trim, and test simple maneuvers. During the first semester of 2007 a set of tests took place in a reservoir in the Douro river, with a maximum depth of 20 meters and about 200 meters wide. These tests allowed the fine tuning of motion control parameters as well as the final adjustments on the acoustic navigation system.

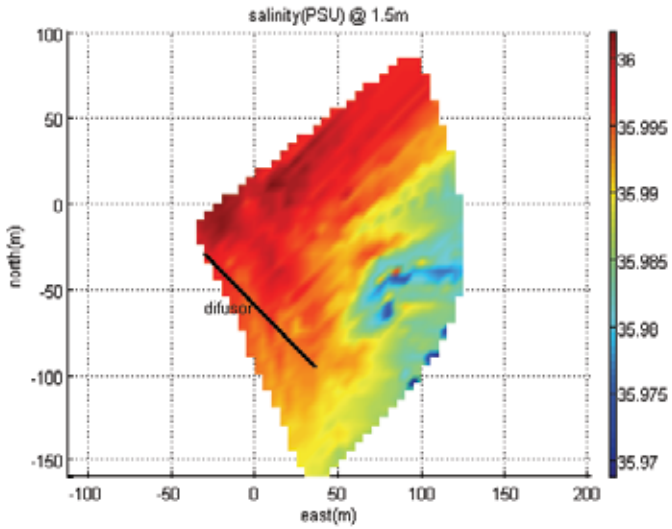


Fig. 7. Salinity map close to the sewage outfall diffuser.

MARES was first demonstrated at sea in November of 2007. This demonstration mission took place in the neighbourhood of a sewage outfall located 2 km off the Portuguese coast at Foz do Arelho. MARES was equipped with a Seabird Fastcat 49 CTD and collected 16 samples/second of CTD data for about one hour. Upon vehicle recovery, CTD data was analyzed to infer the location of the sewage plume in the vicinity of the diffuser. Fig. 7 shows a salinity map produced from CTD data, and although the salinity signature was very weak, it was also very consistent. This demonstrated the potential for detecting minute anomalies with the onboard CTD, which was the main objective for this mission at sea.

The success of the demonstration mission at sea proved that the initial requirements and the design decisions contributed to the development of an operational vehicle that can be effectively used in real application scenarios. One of the major advantages of the MARES AUV, when compared with other AUVs of similar size, is the ability to independently control the motion in the vertical and in the horizontal planes. This allows for some new primitives of motion, such as commanding the vehicle to be completely motionless in the water column (for example, waiting for some triggering event), or diving and emerging vertically, which greatly simplifies its launching and recovery.

7. References

Crowell, J. (2006), Small AUV for Hydrographic Applications, *Proc. MTS/IEEE Oceans'06*, Boston, USA, Sept. 2006

Cruz, N.; Madureira, L., Matos, A.; Pereira, F. (2001), A Versatile Acoustic Beacon for Navigation and Remote Tracking of Multiple Underwater Vehicles, *Proc. MTS/IEEE Oceans'01*, Honolulu, HI, USA, Nov. 2001

Cruz, N.; Matos, A. (2008), The MARES AUV – A Modular Autonomous Robot for Environment Sampling, *Proc. MTS/IEEE Oceans'08 Quebec*, Quebec, Canada, Sept. 2008

Fossen, T. (1994). *Guidance and Control of Ocean Vehicles,* John Wiley & Sons Ltd., ISBN 0471941131

Gelb, A. (1974). *Applied Optimal Estimation,* MIT Press, ISBN 0471941131

Griffiths, G. (2003), *Technology and Applications of Autonomous Underwater Vehicles*, Taylor and Francis, ISBN 0415301548

Healey, A.; Lienard, D. (1993), Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles, *IEEE Jornal of Oceanic Engineering*, vol. 18, no. 3, July 1993

Matos, A.; Cruz, N.; Pereira, F. (2003), Post Mission Trajectory Smoothing for the Isurus AUV, *Proc. MTS/IEEE Oceans'03*, San Diego, CA, USA, Sept. 2003

Matos, A.; Cruz, N.; Pereira, F. (1999), Development and Implementation of a Low-Cost LBL Navigation System for an AUV, *Proc. MTS/IEEE Oceans'99*, Seattle, WA, USA, Sept. 1999

Vaganay, J.; Leonard, J.; Bellingham, J. (2006), Outlier Rejection for Autonomous Acoustic Navigation, *Proc. IEEE Int. Conf. on Robotics and Automation*, Minneapolis, MN, USA, April 1996

von Alt, C.; Allen, B.; Austin, T.; Stokey, R. (1994), Remote Environmental Measuring Units, *Proc. IEEE Symp. AUV Techn. AUV'94*, Cambridge, MA, USA, July 1994

18

Identification of Underwater Vehicles for the Purpose of Autopilot Tuning

Nikola Mišković, Zoran Vukić & Matko Barišić
University of Zagreb, Faculty of Electrical Engineering and Computing
Croatia

1. Introduction

Underwater vehicles (UVs) lately found their use in many activities such as underwater mapping, habitat exploration, different types of inspections (underwater cables, dams, ship hulls), rescue missions and many others. All of these applications speeded up the research related to modeling and control of UVs. Modeling of UVs is important because many applications demand that the mission is carefully planned in a simulated environment. Control, on the other hand, is essential if higher level tasks are to be performed effectively.
Finding a mathematical model describing an underwater vehicle dynamics can be a tedious task. The greatest problem is the complexity of the rigid body dynamics which is augmented by additional forces and effects which appear in fluid. The hydrodynamics observe this problem with great care, paying extra attention to dependencies between different variables. Another problem are the couplings that appear due to motion in different degrees of freedom simultaneously. For the control purposes, most of these dependencies are often neglected in order to obtain a simple model which can later be used for designing autopilots.
This chapter deals with methods for obtaining a precise enough mathematical model, which can be used for control purposes, using cheap and commercially available sensors such as cameras or compasses. First section deals with description of a full mathematical model of an underwater vehicle, starting with actuators (thrusters) and methods of determining their static characteristics. The section is followed by actuator allocation where some common configurations and allocation matrices are mentioned. Kinematic model is briefly addressed and dynamic model is presented in its full form. The coupling effects are observed in the horizontal plane and a short methodology of determining dominant parameters in the coupled model is given. For the uncoupled case, two model equations are taken into account: linear and nonlinear. In section two, three vision-based data acquisition methods are presented. Here it is explained which equipment is needed for making a laboratory apparatus for data acquisition, and the process of data acquisition via image analysis is presented. The following section deals with the identification algorithms which use the data obtained via the vision-based methods. Here we present in short the least-squares method (used for determining the coupled model), open loop, zig-zag methods and identification based on the self-oscillations. All of these methods are followed with results obtained either from real vehicles or simulations.

2. Mathematical models of underwater vehicles

In order to define the full mathematical model of an underwater vehicle (UV) we will use the terminology adopted from Fossen (1994). Vector of positions and angles of an underwater vehicle $\boldsymbol{\eta} = [x \quad y \quad z \quad \varphi \quad \theta \quad \psi]$ is defined in the Earth-fixed coordinate frame $\{E\}$ and vector of linear and angular velocities $\boldsymbol{\nu} = [u \quad v \quad w \quad p \quad q \quad r]$ (surge, sway, heave, roll, pitch and yaw velocity, respectively) is defined in a body-fixed $\{B\}$ coordinate frame, see Fig. 1. Vector $\boldsymbol{\tau} = [X \quad Y \quad Z \quad P \quad Q \quad R]$ represent the external forces that act on the vehicle, vector $\boldsymbol{\tau}^i$ are commanded thrusts for each actuator and $\boldsymbol{n}^i$ are commanded inputs for the actuators themselves. Here we make an assumption that the vehicle is actuated by thruster force, even though other actuator types possible and appear in practice. Using this notation, the complete mathematical model can be represented with Fig. 2. In the following sections, all parts of the model will be described.

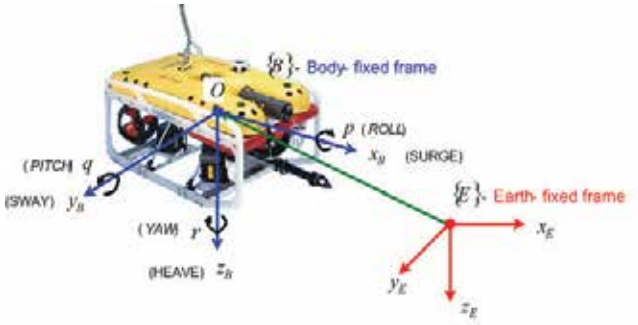


Fig. 1. Body-fixed and Earth-fixed coordinate frames (taken from Omerdic (2004))

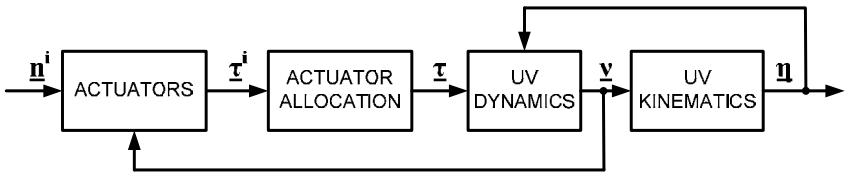


Fig. 2. Scheme of a complete mathematical model

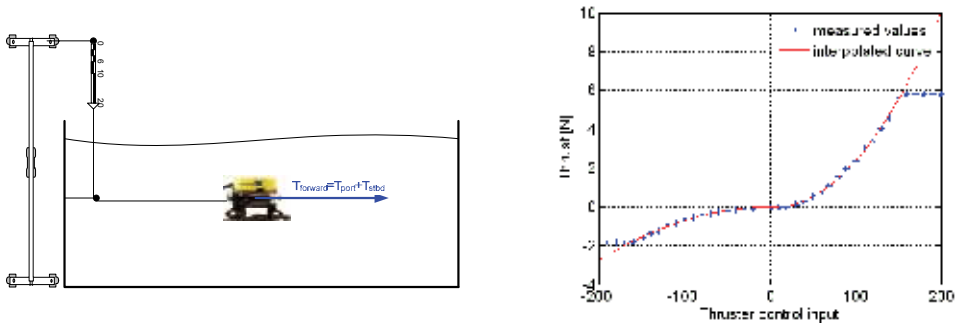
2.1 Actuators

In addition to thrusters as UV actuators, rudders, fins etc. appear in practice also. Here we will limit our discussion on propulsors. According to Fossen (1994), force τ that is exerted by a thruster can be described using a bilinear model, $\tau = b_1|n|n - b_2|n|v$, where b_1 and b_2 are positive constants. This model has revolution rate n as input and vehicle's forward speed v as an additional variable. A simpler model which appears in literature is a model that neglects forward speed, and is given in a form $\tau = a|n|n + bn$. This model is more applicable in practice especially at low speeds. Further simplification gives that linear part of the model can also be neglected, i.e. $b = 0$.

However, the force exerted by thrusters is rarely the same when the propulsor is rotating in both directions. This is why a more complex model (1) should be used where sub indices f and b denote 'forward' and 'backward', and super index i stands for a specific thruster.

$$\tau^i = \begin{cases} a_f |n^i| n^i + b_f n^i, & n^i > 0 \\ a_b |n^i| n^i + b_b n^i, & n^i < 0 \end{cases} \tag{1}$$

Determining the static characteristic of a thruster, i.e. the relation between the exerted thrust and the thruster control signal is called thruster mapping. The procedure consists in exciting the vehicle causing vehicle motion in such a way that the pull-force of the vehicle can be recorder by a dynamometer, as shown in Fig. 3a). An example of thruster mapping results is shown in Fig. 3b) where a VideoRay ROV (two horizontal thrusters and one vertical) is used as a case study. In Fig. 3. dots represent measured values and the full line gives the approximated curve.



a) b)

Fig. 3. Thruster mapping a) experiment and b) results for VideoRay ROV

2.2 Actuator allocation

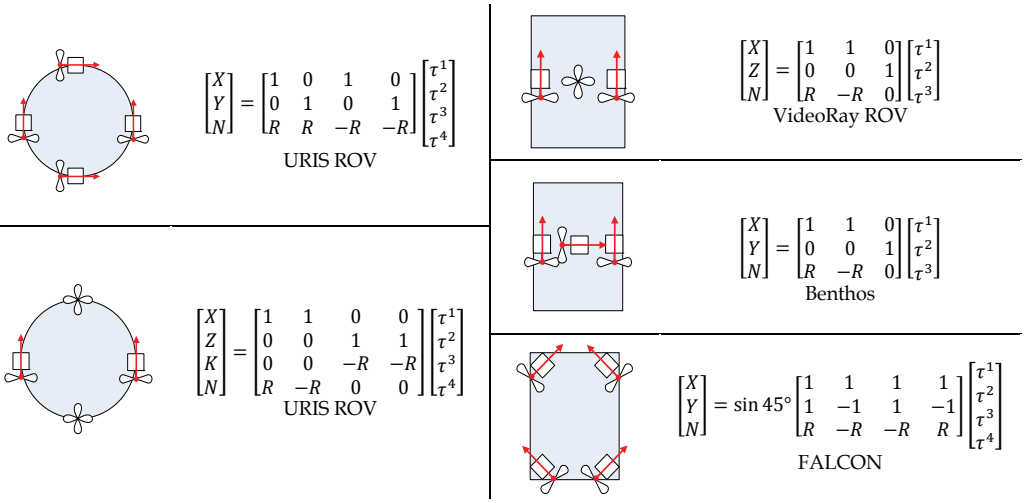


Table 1. Some actuator configurations found in practice

Actuator allocation is a linear connection between the space of actuator forces (described with vector $\boldsymbol{\tau}^i$) and the space of vehicle's forces and moments (described with vector $\boldsymbol{\tau}$). The matrix which describes this link is called the allocation matrix and it depends on the number of available actuators and their topology. Table 1. gives some topologies and actuator allocations which can be found in real vehicles.

2.3 UV kinematics

Kinematic model gives the relation between the speeds $\boldsymbol{v}$ in a body-fixed coordinate frame $\{B\}$ and first derivatives of positions and angles $\boldsymbol{\eta}$ in an Earth-fixed coordinate system $\{E\}$. A full set of kinematic equations is given with (2).

$$\begin{bmatrix}\dot{\boldsymbol{\eta}}_1\\ \dot{\boldsymbol{\eta}}_2\end{bmatrix}=\begin{bmatrix}\boldsymbol{J}_1(\boldsymbol{\eta}_2) & \boldsymbol{0}_{3x3}\\ \boldsymbol{0}_{3x3} & \boldsymbol{J}_2(\boldsymbol{\eta}_2)\end{bmatrix}\begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2\end{bmatrix}$$

$$\boldsymbol{J}_1(\boldsymbol{\eta}_2)=\begin{bmatrix}\cos\psi\cos\theta & -\sin\psi\cos\varphi+\cos\psi\sin\theta\sin\varphi & \sin\psi\sin\varphi+\cos\psi\sin\theta\cos\varphi\\ \sin\psi\cos\theta & \cos\psi\cos\varphi+\sin\psi\sin\theta\sin\varphi & -\cos\psi\sin\varphi+\sin\psi\sin\theta\cos\varphi\\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi\end{bmatrix} \quad (2)$$

$$\boldsymbol{J}_2(\boldsymbol{\eta}_2)=\begin{bmatrix}1 & \sin\varphi\tan\theta & \cos\varphi\tan\theta\\ 0 & \cos\theta & \sin\varphi\\ 0 & \frac{\sin\varphi}{\cos\theta} & \frac{\cos\varphi}{\cos\theta}\end{bmatrix}$$

2.4 UV dynamic model

Dynamic mathematical model of underwater vehicles is coupled and nonlinear. General model equation is given with (3).

$$\boldsymbol{M}\dot{\boldsymbol{v}}+\boldsymbol{C}(\boldsymbol{v})\boldsymbol{v}+\boldsymbol{D}(\boldsymbol{v})\boldsymbol{v}+\boldsymbol{g}(\boldsymbol{\eta})=\boldsymbol{\tau}+\boldsymbol{\tau_d} \quad (3)$$

Matrix $\boldsymbol{M}=\boldsymbol{M_{RB}}+\boldsymbol{M_A}$ presents the sum of rigid body and added mass matrices, matrix $\boldsymbol{D}(\boldsymbol{v})$ is drag (usually diagonal and has linear and quadratic terms), matrix $\boldsymbol{C}(\boldsymbol{v})=\boldsymbol{C_{RB}}(\boldsymbol{v})+\boldsymbol{C_A}(\boldsymbol{v})$ is a sum of Coriolis forces rigid body and added mass matrices, while vector $\boldsymbol{g}(\boldsymbol{\eta})$ contains gravitational and lift forces. Vector $\boldsymbol{\tau}$ contains external forces and moments acting upon the underwater vehicle and $\boldsymbol{\tau_d}$ is the disturbance vector.

From here on we will assume that the only controllable degrees of freedom are surge, yaw and heave, and that sway can appear due to coupling. These assumptions do not limit the applicability of the proposed methods but only simplify them.

Coupled model in the horizontal plane

$$\begin{bmatrix}m-X_{\dot u} & -X_{\dot v} & -my_G\\ -X_{\dot v} & m-Y_{\dot v} & mx_G\\ -my_G & mx_G & I_z-N_{\dot r}\end{bmatrix}\begin{bmatrix}\dot u\\ \dot v\\ \dot r\end{bmatrix}+\begin{bmatrix}0 & 0 & -m(x_Gr+v)+Y_{\dot v}v\\ 0 & 0 & -m(y_Gr-u)-X_{\dot u}u\\ m(x_Gr+v)-Y_{\dot v}v & m(y_Gr-u)+X_{\dot u}u & 0\end{bmatrix}\begin{bmatrix}u\\ v\\ r\end{bmatrix}-\begin{bmatrix}X_u+X_{u|u|}|u| & 0 & 0\\ 0 & Y_v+Y_{v|v|}|v| & 0\\ 0 & 0 & N_r+N_{r|r|}|r|\end{bmatrix}\begin{bmatrix}u\\ v\\ r\end{bmatrix}=\begin{bmatrix}X\\ Y\\ N\end{bmatrix} \quad (4)$$

The reason why the coupled model is limited to the horizontal plane is the simplicity – underwater vehicles can be trimmed in such a way that the heave motion is not affected by the motions in other DOFs. It can be seen from equation (4) that in general case, yaw motion depends on $\dot v+ur$, $\dot u-vr$, uv and N; surge motion depends on $\dot r$, $\dot u$, r^2, rv and X; and sway motion depends on $\dot r$, $\dot u$, r^2 and ur. It is not possible to make general conclusions on which of the parameters can be further neglected - identification has to be performed for each vehicle separately in order to determine the dominant coefficients.

An example of deciding on the model parameters for the coupled model is shown on results for the VideoRay ROV. The correlation between all possible coupling parameters for the three DOFs has been determined and is shown in Fig. 4. The data were obtained using a vision-based method described in Section 2. The terms with the greatest percentage are the dominant ones.

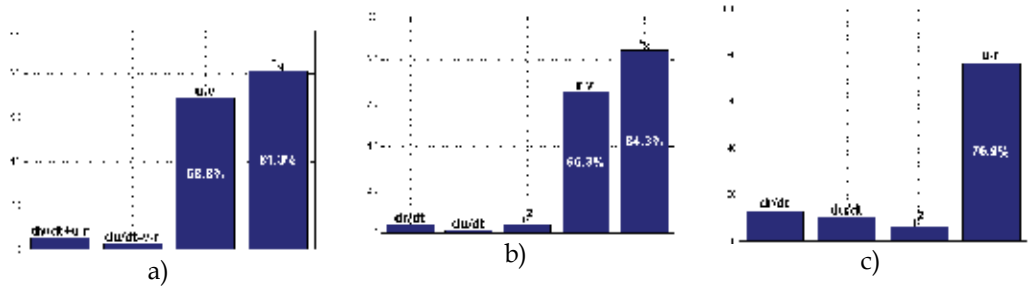


Fig. 4. Correlation coefficients for a) yaw, b) surge and c) sway motion, see Ljung (1999)

These figures let us conclude that parameter $X_{\dot{v}}$ from the added mass matrix is negligible (i.e. the added mass matrix is diagonal) and that the centre of buoyancy is practically equivalent to the centre of gravity ($x_G = y_G = 0$) in the horizontal plane. Both of these statements are true for micro-ROVs, Mišković et al. (2007a). Details on the identified coupled model can be found in Section 4.

Uncoupled model

Additional simplifications can be introduced if the vehicle is trimmed in such a way that roll and pitch are negligible while the vehicle is moving in other controllable degrees of freedom. In order to neglect the coupling due to the Coriolis' forces, it can be assumed that the vehicle is moving at low speed. Using these two assumptions, the coupling can be completely omitted leaving the equation (5) for surge (where $q(t)$ is surge speed u) and yaw ($q(t)$ is yaw rate r) degree of freedom and equation (6) for heave degree of freedom (where $q(t)$ is heave speed w).

$$\alpha\dot{q}(t) + \beta\big(q(t)\big)q(t) = \tau(t) \tag{5}$$

$$\alpha\dot{q}(t) + \beta\big(q(t)\big)q(t) = \delta + \tau(t) \tag{6}$$

In both models, parameter $\tau(t)$ is the excitation force (e.g. surge force X, yaw moment N). Parameter δ can either be external disturbance (in the case of yaw model) or a vehicle physical parameter such as difference between weight and buoyancy (in the case of heave model). This model gives an uncoupled dynamic model of underwater vehicles.

While describing marine vehicle dynamics, usually two models are used based on the drag: the linear one, which has a constant drag coefficient β_L, and the nonlinear one, whose drag coefficient is linear $\beta_N|q(t)|$, Fossen (1994), Caccia et al. (2000), Ridao et al (2004). Linear model is usually used at low speeds, where higher order drag terms can be neglected. This is usually the case when the vehicle is being dynamically positioned. The nonlinear model is suitable for applications where the vehicle is moving at higher speed, i.e. in the cases when the vehicle is in motion.

$$\beta\big(q(t)\big) = \begin{cases} \beta_L & \text{, linear model} \\ \beta_N|q(t)| & \text{, nonlinear model} \end{cases} \tag{7}$$

3. Vision-based data acquisition techniques

In order to determine a mathematical model of an underwater vehicle, which is suitable for control purposes, a great number of sensors can be used to acquire necessary data, e.g. inertial measurement units, positioning systems etc. One of the cheapest and simplest methods for determining the mathematical model parameters is using vision-based techniques to determine the vehicle position. Once the position has been determined, the data can be used to calculate higher order derivatives and thus dynamic model parameters.

3.1 Laboratory apparatus

An interesting vision-based laboratory apparatus used for UV parameter identification was introduced by Ridao et al. (2004). It was based on using a floor pattern at the bottom of the laboratory pool. The apparatus was used with URIS underwater vehicle, which is equipped with a down facing camera - it was placed in a swimming pool with a specifically "coded" floor pattern, see Fig. 5a). Using the image analysis on the frames obtained from the onboard camera, the vehicle position can be uniquely determined.

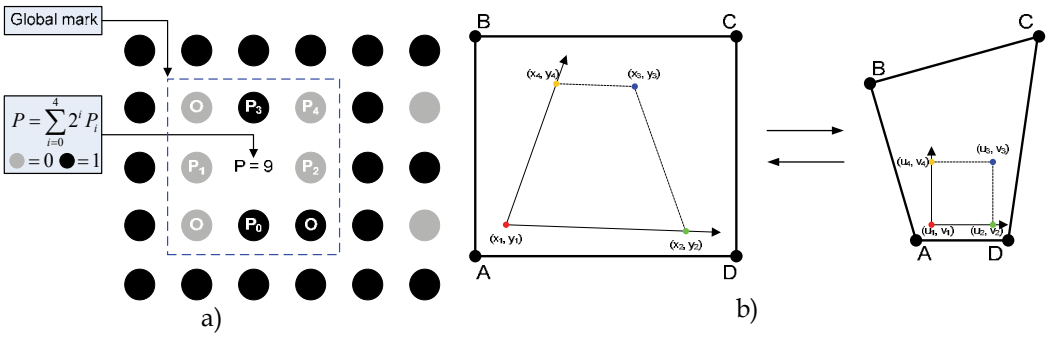


Fig. 5. a) Floor pattern used for URIS UV identification, b) mapping of the swimming pool from a perspective to orthogonal view

The pattern consists of black and grey dots on a white surface. Places without dots are surrounded with global marks. Each global mark is unique and can be decoded based on the combination of the black and grey dots marked with P. In addition to that, dots marked with O are used to determine the orientation of the vehicle. After using the decoding algorithm, vehicle's position within the laboratory pool can be determined. This data is then used for determining the dynamic model of the vehicle. For details on the method, the reader is referred to Ridao et al. (2004) at references within. Even though this method is innovative; the downside is the complexity of the algorithm used for determining the position of the vehicle.

Another approach is to use an external camera placed next to the pool. This way the vehicle can be detected within subsequent frames and its model can be determined. In Chen (2008) the method that is used is based on placing a camera in such a way that the perspective view of the pool is obtained. A schematic representation is shown in Fig. 5b) where points A, B, C and D mark the edges of a frame and the coordinate system (with (x, y) points) is view of the pool within the frame. In order to get the orthogonal projection of the pool (such that the coordinate system is orthogonal) a linear transformation has to be performed – points (x_i, y_i) have to be translated into points (u_i, v_i). This operation will distort the frame so the "upper"

part of the pool has worse resolution than the "lower" part. In order to obtain satisfactory identification results, the camera should be placed in such a way that the frame segment with the worst resolution provides give good results.
The method that the authors have used is based on placing a webcam directly above the swimming pool like in Fig. 6a, Miskovic et al. (2007a). This way the orthogonalization of the pool view is avoided and the algorithm itself is simpler. It should be mentioned that this method can be used for identification of mathematical models of surface marine vessels and underwater vehicles. In order to ensure easier detection of a vehicle within the camera view, a marker is placed on top of the ROV so that its position and orientation within the camera frame could easily be extracted from the recorded video (Fig. 6b). Since the depth cannot be detected with a camera positioned like this, the identification procedure can be performed only in the horizontal plane considering surge, yaw and sway.

a) b)

Fig. 6. a) Laboratory setup for marine vehicle model identification and b) a frame from the webcam placed above the pool

3.2 Data acquisition

The scheme of data acquisition system is shown in Fig. 7. The 'Synchronization' block is used to ensure that a frame is recorded and that control signals are sent once every sample time (100 ms). Once the synchronization is achieved, the procedure can be described as follows:

- Acquire an RGB image from the camera and separate it to a red, green and blue component;
- Transfer the image to a binary equivalent where detection of the red color results in a logical 1 (white) and everything else results in a logical 0 (black). The result of this operation is shown in Fig. 8a.
- Find the centroid of the group of white pixels - this is the position of the ROV within the camera frame.
- Find the orientation of the group of white pixels - this is the orientation of the ROV within the camera frame. The result of this analysis is shown in Fig. 8b where the original camera image is augmented with ROV's position (green circle) and orientation (blue line).
- Perform inverse kinematics on the data using, to obtain linear and angular speeds that are required for model identification.

An example of obtained velocities using camera data is shown in Fig. 8c. Raw data from camera are naturally noisy, therefore they should be filtered.

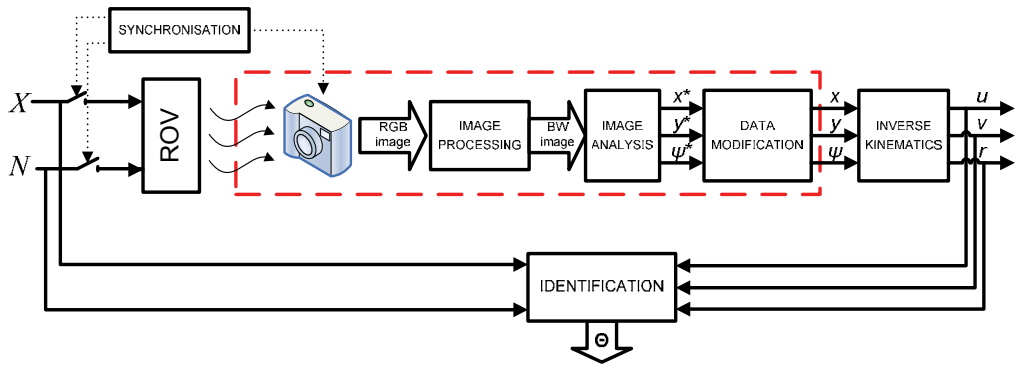


Fig. 7. Video-based data acquisition scheme

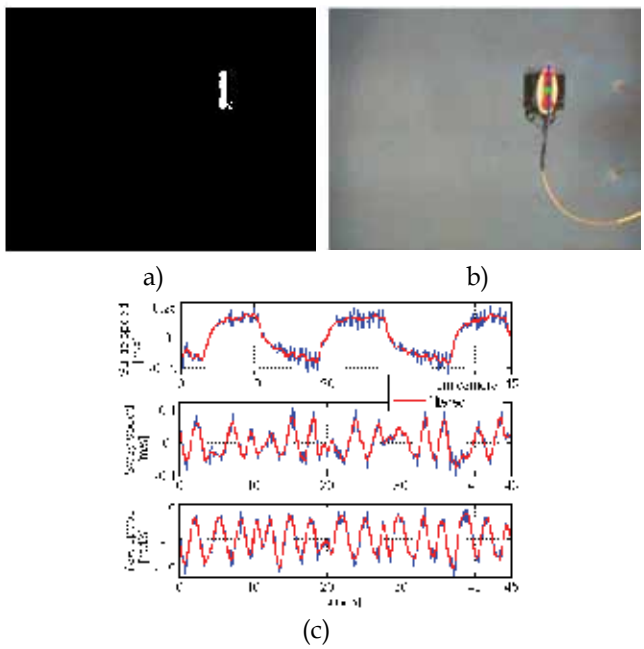


Fig. 8. a) Binary image; b) image augmented with calculated position and orientation; c) an example of obtained velocities

4. UV identification techniques and results

4.1 Least-squares identification method

The least-squares method is a classical identification method which is suitable for determining parameters of complex models, Ljung (1999). The authors have used this method to determine the coupled mathematical model. The main disadvantage of this method is that the observed vehicle has to be persistently excited in all directions for which the model is to be determined. Having this in mind, the VideoRay ROV was driven in a so-called "S-maneuver" in which the vehicle is moving forward-backward and rotating left to right at the same time. This way all couplings in the model are identified.

Based on the coupled model presented and derived in Section 2.4, equations (8), (9) and (10) can be set for surge, yaw and sway motion, respectively, where $\alpha_1 = \frac{N_r}{I_Z - N_{\dot{r}}}$, $\alpha_2 = \frac{1}{I_Z - N_{\dot{r}}}$, $\alpha_3 = -\frac{X_{\dot{u}} - Y_{\dot{v}}}{I_Z - N_{\dot{r}}}$, $\beta_1 = \frac{X_u}{m - X_{\dot{u}}}$, $\beta_2 = \frac{1}{m - X_{\dot{u}}}$, $\beta_3 = \frac{m - Y_{\dot{v}}}{m - X_{\dot{u}}}$, $\gamma_1 = \frac{Y_v}{m - Y_{\dot{v}}}$ and $\gamma_2 = -\frac{m - X_{\dot{u}}}{m - Y_{\dot{v}}}$.

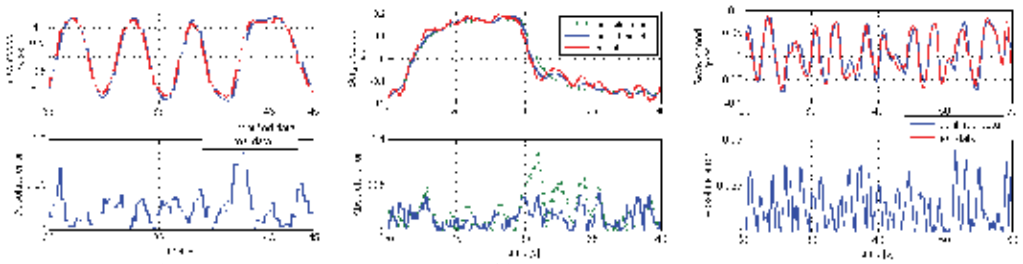
$$\dot{r} = \alpha_1 r + \alpha_2 N + \alpha_3 uv \tag{8}$$

$$\dot{u} = \beta_1 u + \beta_2 X + \beta_3 rv \tag{9}$$

$$\dot{v} = \gamma_1 v + \gamma_2 ur \tag{10}$$

The identified parameters β_3 and γ_3 should be inverse and reciprocal. The identification results show that both parameters are close to 1 so it can be approximated that $\beta_3 = -\gamma_2 = 1$, i.e. added mass terms in surge and sway direction are equal. As a consequence of this $\alpha_3 = 0$, i.e. yaw motion is not coupled to other two motions. Finally the identified model of the VideoRay ROV can be shown in a matrix form using (11). Details can be found in Mišković et al. (2007a). The validation results, which also give comparison between the coupled and uncoupled identified model, are shown in Fig. 9.

$$\begin{bmatrix} 8.24 & 0 & 0 \\ 0 & 8.24 & 0 \\ 0 & 0 & 0.04 \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} 0 & 0 & -8.24v \\ 0 & 0 & 8.24u \\ 8.24v & -8.24u & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} - \begin{bmatrix} -4.43 & 0 & 0 \\ 0 & -14.96 & 0 \\ 0 & 0 & -0.1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} = \begin{bmatrix} X \\ 0 \\ N \end{bmatrix} \tag{11}$$



a) b) c)

Fig. 9. Validation results for a) yaw, b) surge and c) sway motion

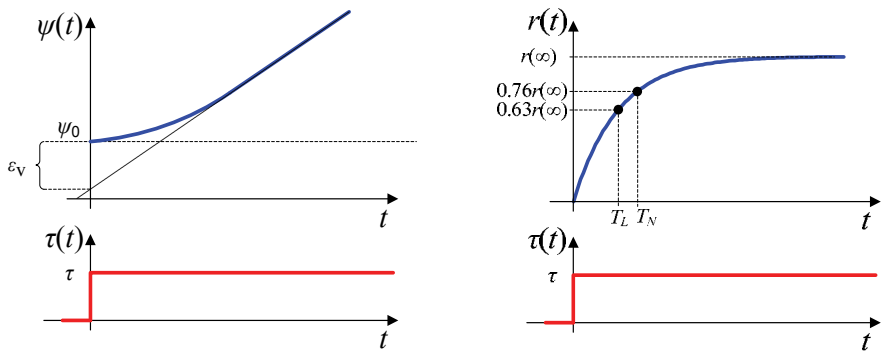
4.2 Identification based on open-loop step response

Open loop identification methods are suitable for laboratory purposes. They are often time consuming since a great number of experiments should be run in order to obtain satisfactory results. On the other hand, these experiments give precise model parameters – the more experiments, the more precise results.

Two cases are possible:

1. The response is constantly rising. This is the case with astatic systems, i.e. when the observed variables are "positions" like heading, depth etc. In this case the SISO system response has a form shown in Fig. 10a.
2. The response has a steady state. This is the case with static systems, i.e. when the observed variables are velocities like heading rate, heave speed etc. In this case the SISO system response has a form shown in Fig. 10b.

For both cases, the drag coefficient can either be constant or linear. From now on we will assume that yaw is the observed degree of freedom.



a) b)

Fig. 10. Open loop step responses for a) astatic and b) static system

Astatic system

In this case we can observe the slope of the steady-state response, $k_{ss} = \lim_{t\to+\infty} \frac{d}{dt}\psi(t)$ and the velocity error, $\varepsilon_v = \lim_{t\to+\infty}[\psi(t) - k_{ss}t]$, Vukić & Kuljača (2005). If a linear model (constant drag coefficient) is assumed, one degree of freedom of an underwater vehicle can be described with (12).

$$\alpha\ddot{\psi}(t) + \beta_L\dot{\psi}(t) = \tau S(t) \tag{12}$$

The solution of this system is given with (13)

$$\psi(t) = \tau\frac{\alpha}{\beta_L^2}\left(e^{-\frac{\beta_L}{\alpha}t} - 1\right) + \frac{\tau}{\beta_L}t + \psi_0 \tag{13}$$

The slope of the steady state response for this case is $k_{ss} = \frac{\tau}{\beta_L}$ and it can be determined from the experiments. Since the applied input is known, constant drag can be precisely calculated depending on the number of experiments. The velocity error for this case is $\varepsilon_v = \psi_0 - \frac{\alpha}{\beta_L^2}\tau$. In other words, step input and initial heading are known a priori, drag coefficient has been determined in the previous step, and velocity error can be determined from the response making it easy to calculate the inertia of the system.

If the system is described as nonlinear, i.e. with linear drag, then the SISO equation is

$$\alpha\ddot{\psi}(t) + \beta_N|\dot{\psi}(t)|\dot{\psi}(t) = \tau S(t) \tag{14}$$

and the response is explicitly given with (15) (see Appendix A for derivation of the term).

$$\psi(t) = \psi_0 + \mathrm{sgn}(\tau)\frac{\alpha}{\beta_N}\left(-\ln 2 + t\frac{\sqrt{\beta_N|\tau|}}{\alpha} + \ln\left|1 + e^{-2t\frac{\sqrt{\beta_N|\tau|}}{\alpha}}\right|\right) \tag{15}$$

The slope of the steady state response for this case is $k_{ss} = \sqrt{\frac{\tau}{\beta_N}}$. Just as in the case before, linear drag coefficient can be calculated easily based on the known parameters. The velocity error for this case is $\varepsilon_v = \psi_0 - \mathrm{sgn}(\tau)\frac{\alpha}{\beta_N}\ln 2$. The result is similar as in the case with constant drag coefficient, only that a constant multiplying term appears. Again, inertia term can be calculated based on the parameters which are known from before.

Static system

In this case we can observe the value of the steady-state response, $K_{ss} = \lim_{t\to+\infty} \frac{d}{dt}\psi(t)$ and some characteristic points of the response. If a linear model (constant drag coefficient) is assumed, one degree of freedom of an underwater vehicle can be described with (16) and the response is explicitly given with (17).

$$\alpha\dot{r}(t) + \beta_L r(t) = \tau S(t) \tag{16}$$

$$r(t) = \frac{\tau}{\beta_L}\left(1 - e^{-\frac{\beta_L}{\alpha}t}\right) \tag{17}$$

The steady state value of the response is clearly $K_{ss} = \frac{\tau}{\beta_L}$. Just as in the astatic case, given enough experimental data, drag can be determined as precisely as needed. The calculation of inertia term is somewhat different than in the astatic case. Here we can use a classical method for determining system's time constant based on the fact that at the time instance $t = T_L = \frac{\alpha}{\beta_L}$ system response achieves around 63% of the steady state value.

$$r(T_L) = \frac{\tau}{\beta_L}(1 - e^{-1}) = r(\infty)(1 - e^{-1}) \approx 0.63r(\infty) \tag{18}$$

Therefore if T_L is determined, based on the known constant drag coefficient, inertia term can be easily calculated.

If the system is described as nonlinear, i.e. with linear drag, then the SISO equation is (19) and the response is explicitly given with (20) (see Appendix A for derivation of the term).

$$\alpha\dot{r}(t) + \beta_N|r(t)|r(t) = \tau S(t) \tag{19}$$

$$r(t) = \sqrt{\frac{|\tau|}{\beta_N}}\tanh\left[\operatorname{sgn}(\tau)\frac{\sqrt{\beta_N|\tau|}}{\alpha}t\right] \tag{20}$$

The steady state value of the response is $K_{ss} = \operatorname{sgn}(\tau)\sqrt{\frac{|\tau|}{\beta_N}}$, and linear drag can be calculated using this value. Similarly as in the case with constant drag coefficient, at the time instance $t = T_N = \frac{\alpha}{\sqrt{\beta_N|\tau|}}$ the system response achieves around 76% of the steady state value.

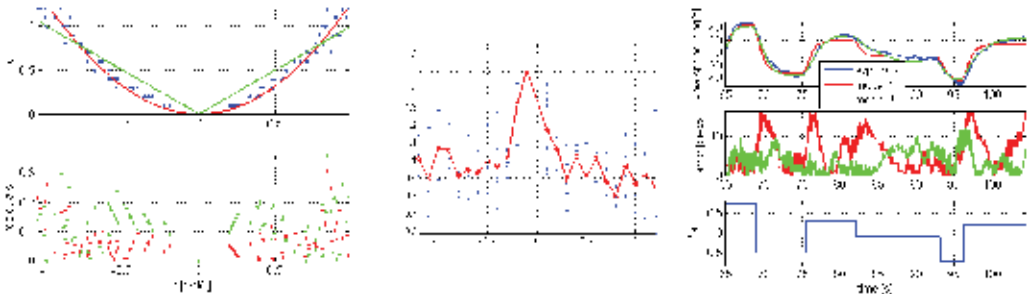
$$r(T_N) = r(\infty)\tanh 1 \approx 0.761r(\infty) \tag{21}$$

Therefore if T_N is determined, based on the known linear drag coefficient, inertia term can be easily calculated. Table 2. gives a short overview of equations for determining model parameters using the open loop experiments.

	linear model	nonlinear model
astatic model	$\alpha = (\psi_0 - \varepsilon_v)\frac{\tau}{k_{ss}^2}, \quad \beta_L = \frac{\tau}{k_{ss}}$	$\alpha = \frac{1}{\ln 2}(\psi_0 - \varepsilon_v)\frac{\tau}{k_{ss}^2}, \quad \beta_N = \frac{\tau}{k_{ss}^2}$
static model	$\alpha = T_L\frac{\tau}{K_{ss}}, \quad \beta_L = \frac{\tau}{K_{ss}}$	$\alpha = T_N\frac{\tau}{K_{ss}}, \quad \beta_N = \frac{\tau}{K_{ss}^2}$

Table 2. An overview of equations for parameter calculation based on open loop step responses

The open-loop identification method was applied to VideoRay Automarine AUV yaw degree of freedom, Stipanov et al. (2007). The case vehicle was developed at the University of Zagreb, Faculty of Electrical Engineering and Computer Science, Laboratory for Underwater Systems and Technologies. The identification results are shown in Fig. 11 and the identified inertia and drag are shown in Table 3. Blue dots show the experimental data, green line shows the fitted nonlinear drag curve while the green line shows the fitted linear drag curve. It is obvious that nonlinear mathematical model describes this vehicle better. This fact was used in determining yaw inertia parameter (see Fig. 11b). Validation results are shown in Fig. 11c where blue line shows the experimental data, red line the linear model and green line the nonlinear model.



a) b) c)

Fig. 11. Experimental results for a) drag and b) yaw calculation, and c) validation results. Green line are results for the fitted linear model, and red line for the nonlinear model

Parameter	α	β_N
	1.018	1.257

Table 3. Identified nonlinear model parameters for VideoRay Automarine Module

4.3 Zig-zag method applied to underwater vehicles

In determining marine surface vehicles' dynamic behavior, zig-zag maneuvers are widely accepted. Zig-zag maneuver is used for designing ship autopilots, i.e. determining yaw motion of a surface vessel, López et al. (2004). The maneuver which is usually run for ships consists of the following steps, while the ship is sailing in advance at a predetermined speed:

- turn the rudder at the maximum speed to the starboard side at 10° (20°)
- when ships course changes by 10deg (20°) from the initial course, turn the rudder to the opposite side (port) at 10deg (20°). After a while, the ship will turn to port.
- when ship course changes by 10deg (20°) from the initial course on the opposite side, turn the rudder again to the starboard side at 10 deg (20°)

The heading and the rudder position should be recorded all the time during the experiment. This algorithm can be simulated as shown in Fig. 12. The response of the zig-zag maneuver are shown in Fig. 13.

The initial assumption for this method is that yaw motion can be described using a simple Nomoto model given with $\frac{\psi}{\delta} = \frac{K}{(Ts+1)s}$ where δ is rudder deflection, ψ heading and K and T parameters which are to be determined. The same model can be applied to underwater

vehicles yaw model, $\alpha\ddot{\psi} + \beta_L\dot{\psi} = N$ - in this case the exciting force is yaw moment. The unknown parameters can be determined by pure integration of the Nomoto model, López et al. (2004).

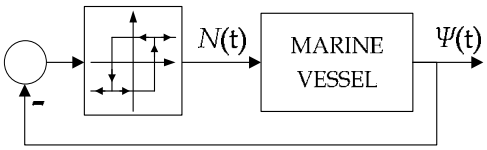
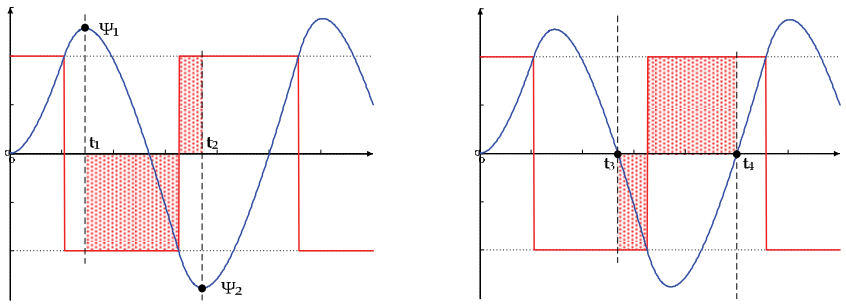


Fig. 12. Simulation scheme for the zig-zag manoeuvre



a) b)

Fig. 13. The zig-zag experiment: integration area for determining a) drag and b) inertia

$$\alpha\dot{r} + \beta_L r = N \tag{22}$$

$$\alpha \int_{t_1}^{t_2} \dot{r}\, dt + \beta_L \int_{t_1}^{t_2} r\, dt = \int_{t_1}^{t_2} N\, dt \tag{23}$$

$$\alpha r|_{t_1}^{t_2} + \beta_L \psi|_{t_1}^{t_2} = \int_{t_1}^{t_2} N\, dt \tag{24}$$

If the integration is performed between the first two time instances when extreme headings appear (the yaw rate at these points equals zero), equation (25) is obtained.

$$\beta_L = \frac{\int_{t_1}^{t_2} N dt}{\psi_2 - \psi_1} \tag{25}$$

If (24) is integrated between two consequent zero crossing point of the heading response, equation (26) is obtained. In this case, yaw rate value at the zero crossing points is needed.

$$\beta_L = \frac{\int_{t_3}^{t_4} N dt}{r_4 - r_3} \tag{26}$$

It is clear that in order to get the two parameters, integration of the control input has to be performed. In Fig. 13 the shaded areas are to be integrated in order to determine inertia and linear drag.

This procedure is practical if linear Nomoto model describes the vessel's dynamic properly. However, if nonlinear terms in the drag appear, the procedure cannot be used. If there is external disturbance present, the Nomoto model should be modified (which makes the procedure much more complex) otherwise the results will be false.

4.4 Identification by use of self-oscillations (IS-O)

The concept of identification by use of self-oscillations is similar to the zig-zag procedure. It was more then 20 years ago when Åström & Hägglund (1984) derived a so called ATV (autotuning variation) method used for system identification, that is simple and appropriate for in situ identification. The method is based on using a relay-feedback to bring the system to self-oscillations. Then Luyben (1987) used this method in chemical industry to identify a transfer function of extremely nonlinear systems (distillation columns). Since then, inducing self-oscillations proved to be a great tool for controller tuning in processes and for process identification, see Li et al. (1991) and Chang & Shen (1992), especially in pharmaceutical industry.

The IS-O method is based on forcing a system into self-oscillations using the same scheme as shown in Fig. 12. In most cases, these oscillations are induced by introducing a relay with hysteresis, but it should be noted that other nonlinear elements can cause the same behavior, Vukić et al. (2003).

Unlike the zig-zag experiment, the IS-O procedure is not based on solving the differential equation which describes the process, but uses magnitudes and frequencies of the induced oscillations to determine system parameters. Having Fig. 12 in mind, the relation between the magnitudes and frequencies of self-oscillations and system parameters can be found through the Goldfarb principle, given with (27) where $G_N(X_m)$ is the describing function of the nonlinear element (relay with hysteresis), X_m is the magnitude of oscillations at the input of the nonlinear element and $G_P(j\omega)$ is the process frequency characteristic.

$$G_P(j\omega) = -\frac{1}{G_N(X_m)} = -\frac{1}{P_N(X_m)+jQ_N(X_m)} \tag{27}$$

Equation (27) can be graphically interpreted as finding intersection points between Nyquist frequency characteristic of the LTI process and an inverse negative describing function of the nonlinear element, see Vukić et al. (2003) and Mišković et al. (2007b). The describing function of the relay with hysteresis is given with (28) and (29) where x_a is half the width of the hysteresis, and C is the relay output.

$$P_N(X_m) = \frac{4C}{\pi X_m}\sqrt{1-\left(\frac{x_a}{X_m}\right)^2} \tag{28}$$

$$Q_N(X_m) = -\frac{4C}{\pi X_m^2}x_a \tag{29}$$

Depending on the number of the unknown process parameters, more different self-oscillations may be obtained. For more details, the reader is referred to Mišković et al. (2007b). In the case of determining inertia and drag, only one experiment is needed.

Let's assume that the process is linear and that it can be described with (5). The self-oscillations will be symmetric and the unknown parameters α and β_L can be found using (30) and (31). If the system is nonlinear, parameter α can be determined using the same equation (30) but parameter β_N is obtained from (32). A more detailed derivation of these expressions can be found in Appendix B. For more details on this, the reader is referred to Mišković et al. (2007c).

$$\alpha = \frac{P_N(X_m)}{\omega^2} \tag{30}$$

$$\beta_L = -\frac{Q_N(X_m)}{\omega} \tag{31}$$

$$\beta_N = -\frac{3\pi}{8}\frac{Q_N(X_m)}{X_m\omega^2} \tag{32}$$

If the system dynamics can be described by using equation (6), i.e. if there is a bias term, than the self-oscillations will not be symmetric. This is the case with heave DOF where there is almost always a difference between weight and buoyancy of the vehicle. If T_H represents the time when relay output is in "high" position, and T_L represents the time when relay output is in "low" position, T_H will differ from T_L. This implies that equations (30), (31), and (32) are not valid. However, based on times T_H and T_L the bias term δ can be determined, using (6) – this way the bias can be compensated for within the controller.

$$\frac{C+\delta}{-(C-\delta)} = \frac{T_H}{T_L} \implies \delta = C\frac{T_H - T_L}{T_H + T_L} \tag{33}$$

This equation can be applied to a general process of n-th order which includes a constant term δ, i.e. can be presented with $f\left(\frac{d^n x(t)}{dt^n}, \cdots, x(t), \delta\right) = \tau(t)$, see Mišković et al. (2008) for details.

The main assumptions that are posed on the self-oscillation method are that the oscillations are symmetric and that higher-order harmonics are negligible in comparison to the first, dominant harmonic. Since these two assumptions are never completely fulfilled in real systems, this method always introduces a slight error in the estimation of the parameters. A detailed analysis on the error which occurs in the application of the method for yaw identification can be found in Mišković et al. (2007c). In the same paper, it is shown that the error will be small if the ratio between the established oscillations and the width of the relay with hysteresis is about 1.5.

The IS-O method was applied to the VideoRay Automarine AUV. The oscillation parameters are shown in Table 4. Since the experiment was performed in a laboratory pool external disturbance is negligible – parameters T_H and T_L are practically the same. The identified parameters of the nonlinear model are shown in Table 5. For more details, the reader is referred to Mišković et al. (2007c).

C	x_a [°]	X_{max} [°]	X_{min} [°]	T_H [s]	T_L [s]
1	20	38	-38.4	2.54	2.34

Tab 4. IS-O method results obtained from the VideoRay Automarine AUV

Parameter	α	β_N
	0.8995	1.2661

Tab. 5. Nonlinear model parameters of the VideoRay Automarine AUV obtained using the IS-O method

The method was also tested on FALCON ROV simulation model, Mišković et al. (2008). Here we only give results of the IS-O method applied to the heave degree of freedom. The response is shown in Fig. 14 (it should be noted that in the simulation model the difference between weight and buoyancy was enlarged so that the infulence of the bias term in (6) would be emphasised). Using (33) it is easy to determine the difference between the weight and buoyancy.

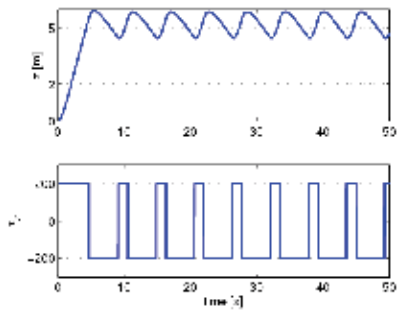


Fig. 14. FALCON ROV simulation model response for IS-O depth experiment

4.5 Zig-zag Vs. IS-O

It is natural to compare these two identification methods because both of them are performed in a closed loop with a nonlinear element, and the system response is identical for both experiments. In addition to that, both methods can be used to determine linear models.

The main difference between the two methods is that the zig-zag method can be used on linear systems only. We have shown that the IS-O method can be applied to nonlinear systems. However, the IS-O method always gives approximate parameter values due to the harmonic linearization assumptions. When the zig-zag method is used, exact parameters are obtained under the assumption that there is no external disturbance. A downside to the zig-zag method is that it uses integration which means that more complex algorithms are needed in comparison to the IS-O method which only uses extreme values of response to calculate the parameters. It should be mentioned that the IS-O requires more than one pair of extreme values so that a median value can be used in order to ensure accuracy.

Fig. 15 shows the error which appears when there is external disturbance present during both experiments. The x-axis values are the percentage ratio between the disturbance and the maximum control value applied during the experiment. The figure clearly shows that the IS-O method is robust to external disturbance, i.e. if this experiment is performed in real conditions, the results will not be exacerbated. This is not the case with the zig-zag experiment. However, by modifying the zig-zag procedure, external disturbance can be taken into account, but the procedure itself becomes more complicated.

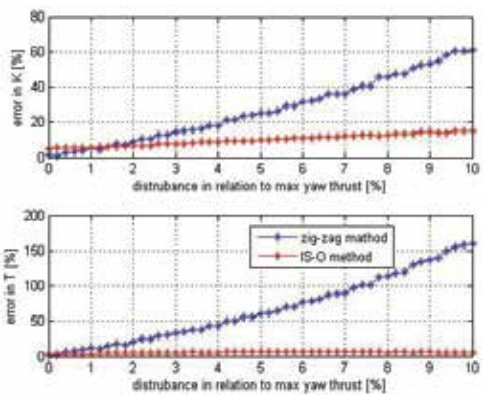


Fig. 15. External disturbance influence on the zig-zag and IS-O method

5. Conclusion

In this chapter the authors have given an overview of some vision-based methods for determining mathematical models of underwater vehicles. The main reason why this type of methods was presented is the commercial availability of the equipment necessary to perform the experiments. Further on, we presented some methods for parameter estimation among which is the self-oscillation identification method. This method has proven to be very applicable in real conditions because of its insensitivity to external disturbance. In addition to that, the method includes simple calculations and is not time-consuming – only a couple of oscillations are enough to estimate the system parameters.

The next step is tuning the autopilots based on the simplified linear or nonlinear mathematical models of underwater vehicles. The IS-O method has already been applied for tuning autopilots of underwater vehicles and results are satisfactory. Further research will be concentrated on tuning higher level controllers using the presented new methods.

Appendix A – Open loop step responses explicit expressions

$$\alpha\dot{r} + \beta_N r|r| = \tau \tag{34}$$

$$\dot{r} = \operatorname{sgn}(\tau)\left(-\frac{\beta_N}{\alpha}r^2 + \frac{1}{\alpha}|\tau|\right)$$

$$\operatorname{sgn}(\tau)\int_0^r \frac{dr}{-\frac{\beta_N}{\alpha}r^2 + \frac{1}{\alpha}|\tau|} = \int_0^t dt$$

$$\operatorname{sgn}(\tau)\frac{\alpha}{\beta_N}\sqrt{\frac{\beta_N}{|\tau|}}\tanh^{-1}\left(r\sqrt{\frac{\beta_N}{|\tau|}}\right) = t$$

$$r = \sqrt{\frac{|\tau|}{\beta_N}}\tanh\left(\operatorname{sgn}(\tau)\frac{\sqrt{\beta_N|\tau|}}{\alpha}t\right) \tag{35}$$

$$\alpha\ddot{\psi} + \beta_N\dot{\psi}|\dot{\psi}| = \tau \tag{36}$$

$$\psi = \int_{\psi_0}^{\psi}\sqrt{\frac{|\tau|}{\beta_N}}\tanh\left(\operatorname{sgn}(\tau)\frac{\sqrt{\beta_N|\tau|}}{\alpha}t\right)dt$$

$$\psi = \sqrt{\frac{|\tau|}{\beta_N}}\operatorname{sgn}(\tau)\frac{\alpha}{\sqrt{\beta_N|\tau|}}\ln\left|\cosh\left(\operatorname{sgn}(\tau)\frac{\sqrt{\beta_N|\tau|}}{\alpha}t\right)\right| + \psi_0$$

$$\psi = \operatorname{sgn}(\tau)\frac{\alpha}{\beta_N}\ln\left|\frac{e^{\frac{\sqrt{\beta_N|\tau|}}{\alpha}t} + e^{-\frac{\sqrt{\beta_N|\tau|}}{\alpha}t}}{2}\right| + \psi_0$$

$$\psi = \operatorname{sgn}(\tau)\frac{\alpha}{\beta_N}\left[-\ln 2 + \frac{\sqrt{\beta_N|\tau|}}{\alpha}t + \ln\left|1 + \mathrm{e}^{-2\frac{\sqrt{\beta_N|\tau|}}{\alpha}t}\right|\right] t + \psi_0 \tag{37}$$

Appendix B – IS-O method equations

Under the assumption that the oscillations of the observed degree of freedom are symmetric, the first and second derivative can be written in the following form:

$$q = X_m \sin(\omega t) \tag{38}$$

$$\dot{q} = \omega X_m \cos(\omega t) = j\omega X_m \sin(\omega t) \tag{39}$$

$$\ddot{q} = -\omega^2 X_m \sin(\omega t) \tag{40}$$

If the observed degree of freedom can be described with a linear dynamic equation in combination with (38), (39) and (40) the following calculations can be done. From this, the final result is obtained and is shown with equations (42), (43) and (47).

$$\alpha\ddot{q} + \beta_L\dot{q} = \tau \tag{41}$$

$$\alpha\{-\omega^2 X_m \sin(\omega t)\} + \beta_L\{j\omega X_m \sin(\omega t)\} = -G_N\{X_m \sin(\omega t)\}$$

$$-\omega^2\alpha + j\omega\beta_L = -P_N - jQ_N$$

$$\alpha = \frac{P_N}{\omega^2} \tag{42}$$

$$\beta_L = -\frac{Q_N}{\omega} \tag{43}$$

$$\alpha\ddot{q} + \beta_N\dot{q}|\dot{q}| = \tau \tag{44}$$

$$\alpha\{-\omega^2 X_m \sin(\omega t)\} + \beta_N\{X_m^2\omega^2 \cos(\omega t)\,|\cos(\omega t)|\} = -G_N\{X_m \sin(\omega t)\}$$

$$\cos(\omega t)\,|\cos(\omega t)| \approx \frac{8}{3\pi}\cos(\omega t) = j\frac{8}{3\pi}\sin(\omega t) \tag{45}$$

$$\alpha\{-\omega^2 X_m \sin(\omega t)\} + \beta_N\left\{X_m^2\omega^2 j\frac{8}{3\pi}\sin(\omega t)\right\} = -G_N\{X_m \sin(\omega t)\}$$

$$-\omega^2\alpha + j\frac{8}{3\pi}\omega^2 X_m\beta_N = -P_N - jQ_N$$

$$\alpha = \frac{P_N}{\omega^2} \tag{46}$$

$$\beta_N = -\frac{3\pi}{8}\frac{Q_N}{X_m\omega^2} \tag{47}$$

6. References

Åström, K. J. & Hägglund, T. (1984). Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica* 1984, *20*, 645.

Caccia, M.; Casalino, G.; Cristi, R. & Veruggio, G. (1998). Acoustic motion estimation and control for an unmanned underwater vehicle in a structured environment. *Control Engineering Practice 6*, (1998), 661-670.

Caccia, M.; Indiveri, G. & Veruggio, G. (2000). Modelling and identification of open-frame variable configuration underwater vehicles. *IEEE Journal of Ocean Engineering*, 25(2), (2000) 227-240

Carreras, M.; Ridao, P.; Garcia, R. & Nicosevici, T. (2003), Vision-based localization of an underwater robot in a structured environment. *IEEE International conference on robotics and automation, ICRA'03*, Taipei, Taiwan

Chang, R. S. & Shen, C. Y. (1992). Derivation of transfer function from relay feedback systems. *Ind. Eng. Chem. Res.* 1992, *31*, 855.

Chen, H.-H. (2008). Vision-based tracking with projective mapping for parameter identification of remotely operated vehicles. *Ocean Engineering*, 35, (2008) 983-994

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*

Li, W.; Eskinat, E. & Luyben, W. L. (1991). An improved autotune identification method. *Ind. Eng. Chem. Res.* 1991, *30*, 1530.

Ljung, L. (1999). *System Identification – Theory for the User*, 2nd ed., Prentice Hall

López, E; Velasco, F. J.; Moyano, E. & Rueda, T. M. (2004). Full-scale manoeuvering trials simulation, *Journal of Maritime Research,* Vol. I. No. 3, 37-50

Luyben, W. L. (1987). Derivation of transfer functions for highly nonlinear distillation columns. *Ind. Eng. Chem. Res.* 1987, *26*, 2490.

Mišković, N.; Vukić, Z. & Barišić, M. (2007a). Identification of coupled mathematical models for underwater vehicles. *Proceedings of the OCEANS'07 Conference, Aberdeen, Scotland*

Mišković, N.; Vukić, Z. & Barišić, M. (2007b). Transfer function identification by using self-oscillations, *Proceedings of the 15th Mediterranean Conference on Control and Applications, Athens, Greece*

Mišković, N.; Vukić, Z. & Omerdić, E. (2008). Control of UUVs based upon mathematical models obtained from self-ocillations experiments", *Proceedings of the NGCUV'08 Conference, Limerick, Ireland*

Mišković, N.; Vukić, Z.; Barišić, M. & Soucacos P. P. (2007c). AUV identification by use of self-oscillations, *Proceedings of the IFAC CAMS'07, Bol, Croatia*

Moreira, L.; Fossen, T. I. & Guedes Soares, C. (2007). Path following control system for a tanker ship model, *Ocean Engineering 34,* (2007), 2074-2085

Omerdic, E. (2004). Thruster Fault Diagnosis and Accommodation for Overactuated Open-Frame Underwater Vehicles. *PhD Thesis,* 2004, University of Wales College, Newport

Ridao, P.; Tiano, A.; El-Fakdi A.; Carreras M. & Zirilli, A. (2004). On the identification of non-linear models of unmanned underwater vehicles, *Control Engineering Practice 12*, (2004), 1483-1499.

Stipanov, M.; Mišković, N.; Vukić, Z. & Barišić, M. (2007). ROV autonomization – yaw identification and Automarine module architecture. *Proceedings of the IFAC CAMS'07, Bol, Croatia*

Vukić, Z. & Kuljača, Lj. (2005). *Automatic Control – Analysis of Linear Systems*, Kigen, Zagreb (in Croatian)

Vukić, Z.; Kuljača, Lj.; Đonlagić, D. & Tešnjak, S. (2003). *Nonlinear Control Systems*, Marcel Dekker, New York

19

Chemical Signal Guided Autonomous Underwater Vehicle

Shuo Pang
Embry-Riddle Aeronautical University
USA

1. Introduction

Olfaction is a long distance sense, which is widely used by animals for foraging or reproductive activities (Dusenbery, 1992; Vickers, 2000; Zimmer & Butman, 2000): homing by Pacific salmon (Hassler & Scholz, 1983), homing by green sea turtles (Lohmann, 1992), foraging by Antarctic procellariiform seabirds (Nevitt, 2000), foraging by lobsters (Basil, 1994), foraging by blue crabs (Wiesburg & Zimmer-Faust, 1994), mating and foraging by insects (Cardé & Mafra-Neto, 1997). Olfaction plays a significant role in natural life of most animals. For some animals, olfactory cues are far more effective than visual or auditory cues in search for objects such as foods and nests (Bell & Tobin, 1982). Although odor sensing is far simpler than vision or hearing, navigation in a chemical diffusion field is still not well understood (Lytridis et al., 2006). Therefore, this powerful primary sense has rarely been used inside the robotics community.

For many military and civilian applications in turbulent fluid flow environment, it would be useful to detect and track a chemical plume to its source. Chemical Plume Tracing (CPT) program, which is sponsored by US Office of Naval Research (ONR), seeks to learn how animals successfully accomplish similar tasks, and to develop algorithms for plume tracing using Autonomous Underwater Vehicles (AUVs). AUVs capable of such chemical plume tracing feats would be of great significance for many applications, e.g., the detection of chemical leaks, locating unexploded ordnance, and locating biologically interesting phenomenon such as thermal vents.

This chapter describes the development and field test of a chemical signal guided REMUS AUV system to find a chemical plume, trace the chemical plume to its source, declare reliably the source location, and map the plume source area after source declaration. The basic idea of the chemical signal guided AUV system is illustrated in Fig. 1. An AUV is constrained to maneuver within a region referred to as the OpArea. Within the OpArea the AUV should search for a specified chemical, for which a binary sensor is available. The mission starts with the AUV searching the OpArea for the chemical plume. A binary sensor outputs 1.0 if the chemical concentration is above threshold or 0.0 if the chemical concentration is below threshold. If above threshold chemical is detected, the AUV should trace the chemical plume to its source and accurately declare the source location. Following the source declaration, additional AUV maneuvers might be desired to acquire additional data, possibly using auxiliary sensors. The plume depicted in Fig. 1 is greatly simplified.

Realistic plumes may meander, are intermittent or patchy distributions of chemical, and do not have a uniformly increasing width as a function of the distance from the chemical source.

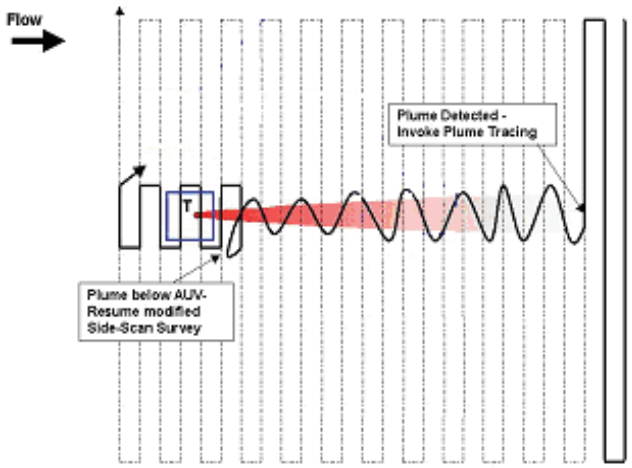


Fig. 1. A prototype CPT mission with post-declaration maneuvering. The depicted plume is a rendition that does not attempt to include intermittency or meander.

A typical vehicle hardware, control, guidance, mapping, and planning architecture for chemical plume tracing are shown in Fig. 2. The figure shows that the assumed inputs to the on-line mapping system are sensed concentration $c(p_v(t_i))$, vehicle location $p_v(t_i)$, and flow velocity $u(t_i) = (u_x(p_v(t_i)); u_y(p_v(t_i)))$ at time t_i. The online planner would optimize a desired vehicle trajectory based the online map. The guidance system outputs heading, speed and depth commands to the controller to achieve the planner's desired trajectory without violating the heading and velocity constraints.

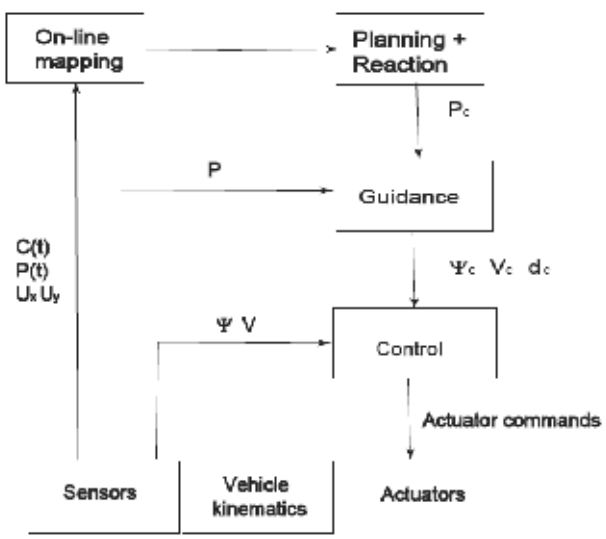


Fig. 2. AUV based Chemical Plume Tracing Architecture.

2. Background

Chemical signal guided search is complicated by the nature of fluid flow and the resulting odor plume characteristics. An initial approach to designing an AUV chemcial plume-tracing strategy might attempt to calculate a concentration gradient. Gradient following based plume navigation algorithms have been proposed for a few biological entities that operate in low Reynolds number environments (Berg, 1990); however, gradient based algorithms are not feasible in environments with medium to high Reynolds numbers (Elkinton et al., 1984; Jones, 1983; Murlis et al, 1992). At low Reynolds numbers, the evolution of the chemical distribution in the flow is dominated by molecular diffusion resulting in a chemical concentration field that is reasonably well-defined by a continuous function with a peak near the source. At medium and high Reynolds numbers, the evolution of the chemical distribution in the flow is turbulence dominated (Shraiman & Siggia, 2000). The flow contains eddying motions of a wide range of sizes that produce a patchy and intermittent distribution of the above threshold chemical (Jones, 1983). For an image of the plume, the gradient is time-varying, steep, and frequently in the wrong direction. Even so, such plume images are not available to the AUV. Due to the rate of spatial and temporal variations in the flow and plume relative to the maneuvering limitations of existing AUV, gradient computation and following is not practical.

If a dense array of sensors were distributed over an area through which a turbulent flow was advecting chemical and the output of each sensor were averaged for a suitably long time (i.e., several minutes), then this average chemical distribution would be Gaussian (Sutton, 1947; Sutton 1953); however, the required dense spatial sampling and long time-averaging makes such an approach inefficient in a turbulence dominated environment (Naeema1 et al., 2007). It is known that the instantaneous chemical distribution will be distinct from the time-averaged plume (Jones, 1983; Murlis et al., 1992). The major differences include: the time-averaged plume is smooth and unimodal while the instantaneous plume is discontinuous and multi-modal; the time-averaged plume is time invariant (assuming ergodicity) while the instantaneous plume is time varying; instantaneous concentrations well-above the time-averaged concentration will be detected much more often than predicted by the Gaussian plume model. Such time-averaged plumes are useful for long-term exposure studies, but are not useful for studies of responses to instantaneously sensed chemical (Murlis et al, 1992). One of the reasons that olfaction is a useful long distance sense is the fact that instantaneous concentrations well above the time-average are available at significant distances from the source (Grasso, 2001). Turbulent diffusion results in filaments of high concentration chemical at significant distances from the source, but also results in high intermittency (Jones, 1983; Murlis et al., 1992; Mylne, 1992). Intermittency increases with down flow distance both due to the meander of the instantaneous plume caused by spatial and temporal variations in the flow and due to the increasing spread with distance of the filaments composing the instantaneous plume. High intermittency and large search areas motivate the need to acquire as much information as is possible from each chemical detection event.

The challenge using chemical signals on AUV is to design effective algorithms to trace the chemical plume and determine the chemical source location even though the chemical source concentration is not know, the advection distance of the detected chemical is unknown, and the flow varies with both location and time.

Various studies have developed biomimetic robotic plume tracing algorithms based on olfactory sensing. The most commonly used olfactory-based navigation algorithms is “chemotaxis" , which was introduced by Berg and Brown (Berg & Brown , 1972; Berg, 1993). This strategy is based on the detection of a concentration difference between two chemical sensors and a steering mechanism toward the direction of higher concentration with a constant moving speed. Chemotaxis-based navigational strategies yield smooth movement trajectories in the environment that the concentration is high enough to ensure its difference measured at two nearby locations is larger than typical fluctuations. Belanger and Willis (Belanger & Willis 1998) presented plume tracing strategies inspired by moth behavior and analyze the performance in a “wind tunnel-type" computer simulation. The main goal of that study was to improve the understanding of moth interaction with an odor stimulus in a wind tunnel. Grasso et al. (Grasso et al. 1996; Grasso, 2001; Grasso, et al., 2000) evaluate biometric strategies and challenge theoretical assumptions of the strategies by implementing biometric strategies on their robot lobster. Li et al. (Li et al., 2001; Li et al., 2006) develop, optimize, and evaluate a counter-turning strategy originally inspired by moth behavior. Vergassola et al. (Vergassola et al., 2007; Martinez 2007) proposed a search algorithm, “infotaxis", based on information and coding theory. For infotaxis, information plays a role similar to concentration in chemotaxis. The infotaxis strategy locally maximizes the expected rate of information gain. Its effciency was demonstrated using a computational model of odor plume propagation and experimental data on mixing flows. Infotactic trajectories feature zigzagging and casting paths similar to those observed in the fight of moths. Spears et al. (Spears et al., 2005; Zarzhitsky et al., 2004) developed a physics-based distributed chemical plume tracing algorithm. The algorithm uses a network of mobile sensing agents that sense the ambient fluid velocity and chemical concentration, and calculate derivatives based on formal principles from the field of fluid mechanics.

The fundamental aspects of these research efforts are sensing the chemical, sensing or estimating the fluid velocity, and generating a sequence of searcher speed and heading commands such that the motion is likely to locate the odor source. Typical maneuvers include: sprinting upflow upon detection, moving crosswind when not detecting, and manipulating the relative orientation of a multiple sensor array either to follow an estimated plume edge or to maintain the maximum mean reading near the central sensor. In each of these articles, the algorithms for generating speed and heading commands use only instantaneous (or filtered) sensor readings.

This chapter extends plume tracing research by presenting a complete strategy for finding a plume, tracing the plume to its source, and maneuvering to accurately declare the source location; and, by presenting results from successful, large-scale, in-water tests of this strategy. The assumptions made herein relative to the chemical and flow are that the chemical is a neutrally buoyant and passive scalar being advected by a turbulent flow. The AUV is capable of sensing position, concentration, and flow velocity. The concentration sensor is used as a binary detector (above or below threshold). We solve the plume-tracing problem in two dimensions. A main motivation for implementing the algorithms in two dimensions is the computational simplification achieved; however, neutral buoyancy of the chemical or stratification of the flow (Stacey, 2000) will often result in a plume of limited vertical extent, which may be approximated as two-dimensional.

3. Behavior based planning method

Chemical signal guided search is a complicated problem. One way to reduce the complexity is to break down the planning problem into a set of simpler subproblems each solvable by simpler actions with an appropriate method to switch between actions. This divide-and-conquer strategy is effective in many planning applications that deal with complex systems. These simpler actions are called behaviors. A behavior is a mapping of sensor inputs to a pattern of motor actions, that accomplishes a single goal within a restricted context. A behavior-based planning (BBP) strategy is an efficient means to navigate an autonomous system in an uncertain environment. To use a set of behaviors to achieve a task a mechanism for coordinating the behaviors is also required.

In the late 1970's and early 1980's, Arbib began to investigate models of animal intelligence from the biological and cognitive sciences point-of-view to gain alternative insight into the design of advanced robotic capabilities (Arbib, 1981). At nearly the same time, Braiten Berg studied methods by which machine intelligence could be evolved by using sensor-motor pairs to design vehicle systems (Braintenberg, 1984). Later, a new generation of AI researchers began exploring the biological sciences in search of new organizing principles and methods of obtaining intelligence. This research resulted in the reactive behavior-based approaches. Brooks' subsumption architecture is the most influential of the purely reactive paradigms. Its basic idea is to describe a complex task by several behaviors, each with simple features (Brooks, 1986). Design of a behavior-based planner includes two significant steps. First, the designer must formulate each reactive behavior quantitatively and implement the behavior as an algorithm. Second, the designer must define and implement a methodology for coordinating the possibly conflicting commands from the different behaviors to achieve good mission performance.

Various coordination approaches have been proposed. For example, each behavior can output a command and a priority. Traditional binary logic can be used to select and output the command with the highest priority. An alternative coordination approach is to use artificial potential fields (Arkin & Murphy 1990). A drawback to either approach is that formulating and coordinating the reactive behaviors requires significant pre-mission simulation and testing. These are ad-hoc processes and may need to be re-addressed each time new behaviors are added or existing behaviors are changed. In some applications, these tuning parameters depend heavily on environmental conditions. Another alternative that has been suggested is to train an artificial neural network (ANN) to perform the behavior coordination (Li et al., 1997). However, this approach would require some mechanism for determining “correct” coordination decisions for each training scenario and would provide no guarantee that all coordination situations are properly trained (Berns et al., 1991). Fuzzy control methods can improve the performance of reactive behavior coordination (Li et al. 1997) by providing a formalism for automatically interpolating between alternative behaviors. Although similar in overall structure, fuzzy control differs from classic feedback control. In fuzzy control, the controller has the same function inputs and outputs as in the feedback control, but internally the control values are computed using techniques from fuzzy logic. Fuzzy controller takes fuzzy state variables, by applying sets of fuzzy rules, produces a set of fuzzy control values. These fuzzy control values are not precise numbers, but rather represent a range of possible values with different weights. Eventually, a decision is made based on the fuzzy control values.

Behavior based design methodologies are bottom-up approaches to the design of an intelligent system. Observed behaviors with simple features are analyzed and synthesized independently. By using these design methodologies, we break down the complicated plume tracking problem into five behaviors. Later in this chapter, we will describe the behaviors and coordination mechanism that were used to solve the problem of chemical plume tracing strategy for an AUV in details. The behaviors were inspired by behaviors observed in biological entities.

4. AUV guidance system

A typical AUV chemical plume tracing system includes an adaptive mission planner (AMP) that rapidly responds to the sensor inputs to generate a trajectory for the AUV to trace the plume. Because the AUV has velocity (<2 m/sec) and heading rate (<10 degree/sec) constraints and the vehicle navigation system has navigation fixes (The vehicle is performing dead-reckoning based on acoustic Doppler data with periodic navigation updates based on data from a long baseline acoustic buoy transponder system. The position updates to the dead-reckoned position based on the LBL data are referred to as navigation fixes), a guidance system is necessary for the AUV to generate heading and speed commands within the constraints to achieve the trajectory desired by the AMP. "Guidance is the action of determining the course, attitude and speed of the vehicle, relative to some reference frame, to be followed by the vehicle" (Fossen, 1994). For the chemical signal guided AUV, the guidance system combined with the AMP decides the best trajectory to be followed by the AUV based on the chemical information and vehicle capability. Although many guidance systems exist for use on the land and air vehicles, there are few, if any systems designed for AUVs (Naeem et al., 2003).

The AUV guidance system is divided into four guidance modes: Go To Point mode, Follow Line mode, Go To Point with Heading mode, and Cage mode. The Go To Point mode is used to drive the AUV from its present location to a destination, without regard to the heading at the destination location. The Follow Line mode is used to track a straight line. The Go To Point with Heading mode is to drive the AUV from a start position and orientation angle to a destination position and orientation angle with the constraint that desired trajectory cannot violate a prespecified minimum turning circle. The Cage mode prevents the vehicle from leaving the operating area or return the vehicle to the operating area if it has left the operating area. To ensure the outputs of the guidance system do not violate the heading rate constraint, the heading commands are filtered before they are sent to the vehicle control unit. In any of these modes, the guidance function will output depth/altitude and earth relative velocity (geographic heading and speed) commands that are within the velocity and heading rate constraints of the AUV. For accurate implementation of the desired trajectory, the guidance system should compensate these commands for the flow vector to produce water relative speed u_c and ground relative heading commands Ψ_g^t:

$$V_f = V_g - F_g \tag{1}$$

$$\Psi_g^t = a\tan 2[(v_f, u_f)^t] \tag{2}$$

$$Vc = \|V_f\| \tag{3}$$

where V_f = (u_f , v_f , w_f) is the water relative AUV velocity, V_g is the ground relative AUV velocity, and F_g is the ground relative flow vector. A superscript indicates a coordinate frame: "t" for geodetic tangent frame or "b" for body frame. The components of vector V_g^t are (u_f , v_f, w_f)^t.

4.1 Go to point

This mode is used to drive the vehicle from its present location to a destination, without regard to the heading at the destination location, e.g., initialize the plume search from a desired point, go to next search region after the vehicle finish searching in the current region, or return to home location after the vehicle finish its mission.

When the guidance system is in Go To Point mode, the output of the system is the geographic heading command

$$\Psi_c = \arctan\left(\frac{y(t) - y_d}{x(t) - x_d}\right) \tag{4}$$

and a constant speed command

$$V_c(t) = v \tag{5}$$

where ($x(t)$, $y(t)$) is the current vehicle position, (x_d, y_d) is the destination location, and v is a predefined constant speed. Note, the heading angle $\Psi_c \in$ [0, 360] is defined in degrees and goes clockwise. When the vehicle is within a radius R of the destination location,

$$\sqrt{(x(t) - x_d)^2 + (y(t) - y_d)^2} < R \tag{6}$$

where R is a predefine value, it is considered to have arrived at the destination location and the guidance system will exit from the Go To Point mode.

This mode is the most robust mode in our guidance system. Because unlike the other modes in our guidance system, the vehicle does not try to follow a precalculated trajectory, instead it calculates its trajectory based on the real time vehicle location information. Therefore, when we have navigation fixes and curvature constrains during the vehicle traveling, the vehicle trajectory is modified accordingly.

4.2 Follow line

Sometimes the vehicle needs to track a straight line, e.g. the vehicle doing a lawn mower search, or the vehicle doing a side scan maneuver after it declares the source location. Given two locations (x_s, y_s) and (x_d, y_d) in the OpArea, we can get a line segment L_{sd} which starts from point (x_s, y_s) and ends at point (x_d, y_d). The Follow Line mode will generate a set of heading and speed commands which will make the vehicle follow the line L_{sd}. The first step to achieve follow line mode is to drive the vehicle to approach the start point (x_s, y_s) while ensuring that the vehicle heading Ψ upon arrival at the start of the line is about the same as the line orientation angle,

$$\alpha_{sd} = \arctan\left(\frac{y_d - y_s}{x_d - x_s}\right) . \tag{7}$$

Here we cannot use Go To Point mode, because it cannot satisfy the heading condition. So we design a new mode Go To Point with Heading to achieve this work. This mode will be discussed later. When the vehicle is within radius R of the start point and within heading angle θ of α_{sd}, the vehicle will begin to follow the line. The corresponding heading command is

$$\psi_c(t) = \begin{cases} \alpha_{sd} + K \times d & d < d_1 \\ \alpha_{sd} + \text{sign}(d) \times 45 & d \geq d_1 \end{cases} \tag{8}$$

where d is the signed distance between the vehicle current position P(t) = ($x(t)$, $y(t)$) and the line L_{sd}, K is a predefined gain, $d_1 = 45/K$, and the sign function is defined

$$\text{sign}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases} \tag{9}$$

Note, the distance d is positive when the vehicle is on the left side of the line L_{sd} (when looking from the start point (x_s, y_s) to the destination point (x_d, y_d)), and negative when the vehicle is on the right side of the line.

The exit condition for this mode is different from Go To Point mode. In Go To Point mode, we exit the mode when the vehicle is within a radius R of the destination location. Here in the Follow Line mode, we can still use this condition. However, since there are some navigation fixes during the vehicle traveling, the vehicle trajectory is not continuous; it contains some jumps in the trajectory. These jumps may happen near the destination point, therefore the vehicle may jump over the destination point without being within radius R, and it will continue following the line until it hits the edge of the OpArea. To prevent this, we need to add one additional exit condition for this mode. When the vehicle pass the destination point in the direction of the line for more than R_L meters we suppose that the vehicle has finished the follow line mode and it exits from this mode. That is, if

$$V \bullet h > R_L \tag{10}$$

where vector $V = [x\text{-}x_d, y\text{-}y_d]$, and $h = [\cos(\alpha_{sd}); \sin(\alpha_{sd})]$ is a unit vector in the direction of the line, then we exit from the follow line mode.

Fig. 3 shows an example of follow line mode. The vehicle is start from position $P_1(x_1, y_1)$. Go To Point with Heading function drives the vehicle to the position $P_2(x_s, y_s)$, which is within a radius R of the start position (x_s, y_s) with heading error less than 15 degrees. Then, the vehicle will follow the line based on the heading command defined in equation (8) until either condition (6) or condition (10) is satisfied.

4.3 Go to point with heading

The goal of this mode is to drive the AUV from a start position and orientation angle to a destination position and orientation angle with the constraint that desired trajectory cannot violate a prespecified minimum turning circle. This guidance mode is significantly more complicated than it first appears. It was proved by Dubins (Dubins, 1957) that this trajectory consists of exactly three path segments. It is either a sequence of *CCC* or *CSC*, where *C* (circle) is an arc of minimal turning radius R_m and *S* (straight line) is a line segment. In our application, we only use the *CSC* trajectory. Even though the *CSC* trajectory sometimes is not the shortest path, it is easy to generate this trajectory, thereby saving computation resources.

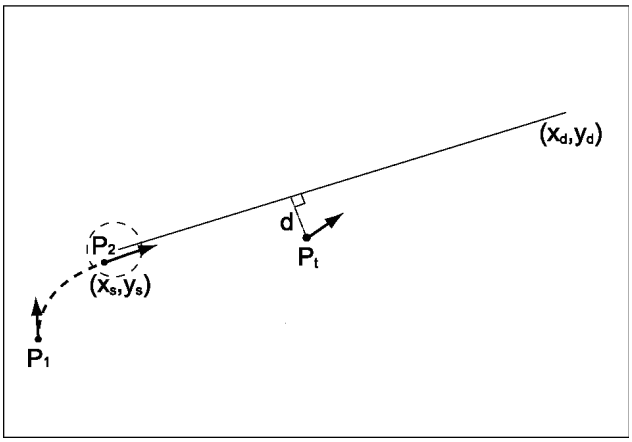


Fig. 3. Definition of variables for the Follow Line mode.

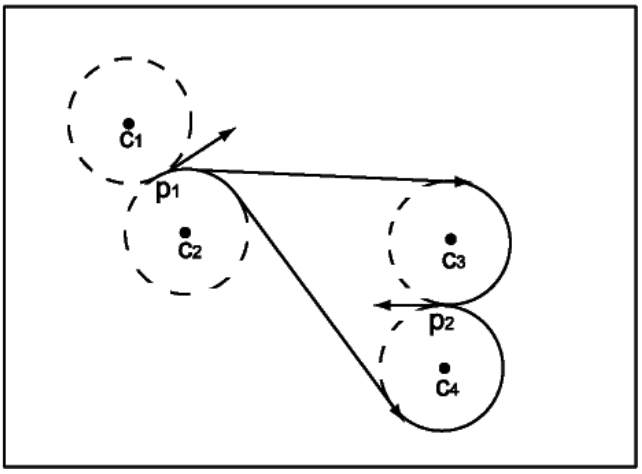


Fig. 4. Depiction of the Go to Point With Heading mode.

Fig. 4 shows an example of Go to Point with Heading mode. The AUV starts from position p_1 with orientation angle θ_1 and should go to position p_2 with orientation angle θ_2. Here we use two unit vectors V_1 and V_2 to represent the start and destination positions and orientation angles. First, we generate four circles C_1, C_2, C_3, C_4, whose radii are the minimal allowed turning radius R_m. The first two circles C_1, C_2 are tangent to V_1 at p_1, C_3, C_4, are tangent to V_2 at p_2. Note that arcs C_1, C_4 are counterclockwise and C_2, C_3 are clockwise. Second, we generate four line segments L_{ij}, where $i=1,2$ and $j=3,4$ (only showing two lines in Fig. 4). Line L_{ij} connects C_i to C_j in a continuous fashion. Now, we have four possible candidate paths, namely, $C_1L_{13}C_3$; $C_1L_{14}C_4$; $C_2L_{23}C_3$; $C_2L_{24}C_4$. Third, we calculate the length for each of the four candidate paths and select the shortest path as the trajectory for the AUV.

4.4 Cage

The Cage mode has two responsibilities related to the safety of the AUV. First, it should prevent the AUV from leaving the operating area or return the AUV to the operating area if it has left the operating area. Second, if the AUV is more than 30 m outside the operating

envelope, then the Cage mode must abort the mission. Aborting the mission in the latter case is straightforward.

When the AUV is outside the OpArea or is near (within 5 m) an edge, we find the outward unit normal $N=[N_e,N_n]$ and the distance δ to the nearest edge. If the AUV is inside the OpArea (i.e., $0<\delta<5$), then the commanded heading that results from the guidance system is modified to remove a portion of its outward component:

$$V = [\cos(\psi_c), \sin(\psi_c)] \tag{11}$$

$$T = V - (1-\delta/5)(V^T N)N \tag{12}$$

$$\psi_c = \text{atan2}(T_1,T_2) \tag{13}$$

where "atan2" is the four quadrant arc tangent function. Therefore, when inside the OpArea, the AUV should not drive itself out of the OpArea; however, a navigation fix could instantaneously change the computed AUV position to be outside of the OpArea. If the AUV is outside the OpArea, then the heading command is $\psi_c = \text{atan2}(-N_e,-N_n)$.

5. Behavior based chemical plume tracing

Fig. 5 displays the behaviors and switching logic used to implement CPT algorithms using BBP. In Fig. 5, *S* and *d* are Boolean variables. The symbols *S* and $\overline{S}$ indicate that the source location has or has not been declared, respectively. The symbol *d* indicates that chemical has been detected. The symbol $\overline{d}$ indicates that the behavior completed without detecting chemical. Prior to source declaration, whenever chemical is detected, the Track-In behavior

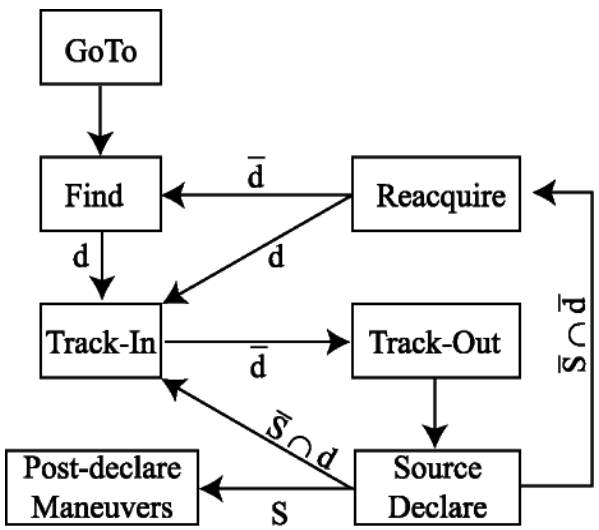


Fig. 5. Behavior Switching Diagram. The symbol *d* denotes a behavior switch that occurs when chemical is detected. The symbol $\overline{d}$ denotes a behavior switch that occurs when chemical is not detected prior to the end of the behavior. *S* indicates that the source location has been declared. $\overline{S}$ indicates that the source location has not been declared.

is triggered. Due to the intermittency caused by the turbulent flow, an instantaneous chemical reading below the detection threshold does not necessarily imply that the AUV is "out of the plume." Therefore, the sequence of behaviors Track-Out, Reacquire, Find is instantiated as the time since the last detection increases. The specific aspects of each behavior and the logic for switching between the behaviors are described in later. The planner is implemented on a PC104 computer that will be referred to as the Adaptive Mission Planner (AMP).

5.1 Go-To behavior

The Go-To behavior is used to drive the vehicle to a desired location. This is used for example at the start of a mission to maneuver the vehicle to a desired starting location. The Go-To behavior directly executes the Go-To guidance command.

5.2 Find behavior

Since there is no prior information about the location of the source, the AUV may be required to search the entire OpArea. Since the odor plume will be downflow from the source, the search is designed to start at the most downflow corner of the OpArea. From this starting location, the AUV should proceed across the flow until it either reaches a boundary of the OpArea or detects chemical. Although the largest component of the commanded velocity is across the flow, there must also be a component either up or down the flow so that the AUV will explore new locations in the OpArea. If chemical is detected, then the behavior switches to Track-In. If the AUV meets the boundary without detecting chemical, then the reaction is described below.

When the AUV arrives at a boundary, four candidate directions are computed as: $\psi_f \pm 90 \pm 20$, where ψ_f is the flow direction in degrees. Of these four candidate directions, the behavior selects the direction that maintains the same sign of the velocity along the boundary and reverses the sign of the velocity perpendicular to the boundary. When none of the four candidates satisfies this condition, then the motion is continued parallel to the boundary until the condition is achieved or another boundary is met. At such a corner, two directions of motion must be changed, and the solution can always be found. When the flow is parallel to a boundary, then this Find strategy results in a billiard ball type of reflection at the OpArea boundary.

5.3 Track-In behavior

Studies described in (Li et al., 2001) show that immediately following a chemical detection, good plume tracking performance is attained by driving at an angle $\beta \in [20,70]$ degree offset relative to upflow. When driving at a nonzero angle β offset relative to upflow and contact with the plume is ultimately lost, the AUV can predict which side of the plume it exited from and perform a counterturn to reacquire the plume. Such counterturning strategies are exhibited in several biological entities. The Track-in behavior implements an engineered version of such a strategy.

Pseudo-code for the Track-In behavior is contained in Table 1. The AMP will stay in Track-In behavior as long as there has been an above threshold concentration sensed in the last λ seconds. While chemical is being detected, AMP adjusts the commanded heading ψ_c to be offset by LHS*β relative to the upflow direction $\psi_u = \psi_f + 180$. In this expression β is a

constant and LHS is a variable that switches based on the relative directions of the AUV and flow. LHS is 1 if we expect the AUV to drive out of the plume from the left side (when looking upflow) of the plume. Otherwise, LHS is –1. Each time chemical is detected, the current AUV position is saved; therefore, when Track-In exits, the last detection point is available and saved in a list named lost_pnts.

```
Behavior::track_in( )
{
   v = v_c ;
   if(odor conc. >= threshold)
   { // Stay in track in
       ψc       = ψf + 180 + LHS*β;
       if (ψv- ψf <180)
        LHS =1;
       else
        LHS =-1;
       T_last = t;
       last_detection_point = position;
   }
   else if((t - T_last) > λ)
   { // Go to track out
    // save last detection point
    lost_pnts[i]=last_detection_point;
    i++;
    return track_out;
    }
   return track_in;
}
```

Table 1. Pseudo Code for Track-In Behavior

As long as the AUV is detecting chemical at least every λ seconds, it will make up flow progress. The actual AUV trajectory will include small angle, counter-turning oscillations relative to the upflow direction. If the AUV fails to detect chemical for λ seconds, then AMP saves the last detection point (at most 6 points are saved) and switches to Track-Out.

5.4 Track-Out behavior

Pseudo-code for the Track-Out behavior is contained in Table 2. When the AMP switches to Track-Out, it has detected chemical slightly more than λ seconds previously; in addition, there will be at least one point on the list of last detection points. Normally, the most recent detection point will be the last one on the list; however, since other behaviors manipulate the list, this is not guaranteed. Also, the variable LHS indicates on which side of the plume the AUV was located when contact with the plume was lost.

The Track-Out behavior attempts both to make progress towards the source (upflow) and to quickly reacquire contact with the plume. To accomplish these two objectives, AMP commands the AUV to go to a point that is L_u meters upflow and L_c meters across the flow from the most upflow point on the list of last detection points. The crossflow direction is

selected so that, if chemical is not detected, the AUV is expected to end up on the opposite side of the plume, since crossing the plume increases the likelihood of detecting chemical. Track-Out ends either when chemical is detected or the AUV arrives at the commanded location. In either case the BBP checks whether it can declare a source location prior to determining the next maneuver. If the source is declared, then post-declaration maneuvering begins. If chemical is detected and the source location cannot be declared, then the behavior switches to Track-In. In this case, the AUV is at a location further up the plume than the previous most upflow detection point. If the AUV arrives at the commanded point without detecting and the source location cannot be declared, then the behavior switches to Reacquire.

```
Behavior:: track_out( )
{
   v = v_c ;
   if(odor conc. >= threshold)
   {
    init = 1;
    S = src_chck(lost_pnts);
    if(S)
       return post_dclr;
    else
       return track_in;
   }
   else
   {
    if(init==1)
    {
     UP = upflow pnt in lost_pnts
     init = 0;
     pnt=UP - L_u*F - L_c*LHS*F_p;
    }
    if(‖veh_pos -pnt‖<R)
    {
      init = 1;
      S = src_chck(lost_pnts);
      if(S)
       return post_dclr;
      else
       return reacquire;
    }
    else
      goto(pnt);
   return track_out;
   }
}
```

Table 2. Pseudo Code for Track-Out Behavior. F is a unit vector in the direction of the flow. F_p is rotated positively by 90 degree relative to F in the horizontal plane. R, L_u, and L_c are positive constants.

```
Behavior::reacquire( )
{
 if(odor conc. < threshold){
   pnt= find_upflow_last_pnt();
   if(n < N_re){
     center_pnt=pnt -
       10(N_re-1-n)/(N_re-1)*F;
     if(bow_tie(center_pnt)== done)
       n++;
     }
    else{
      n = 0;
      remove_pnt_from_list(pnt);
      if(last_pnt_list_is_empty())
        return find;
    }
    return reacquire;
  }
  else{
    n = 0;
    return track_in;
  }
}
```

Table 3. Pseudo Code for Reacquire Behavior

5.5 Reacquire behavior

Pseudo-code for the Reacquire behavior is contained in Table 3. When the AMP switches to Reacquire, it has not detected chemical for several seconds; however, there will be at least one point on the list of last detection points. Also, the variable LHS indicates the side of the plume on which the AUV was when it lost contact with the plume. To switch to the Reacquire behavior, the Track out behavior must have completed without detecting chemical. Therefore, several scenarios could have occurred:

- The AUV could be upflow from the source.
- The AUV could have crossed the (intermittent) plume without detecting chemical.
- If the LHS variable was incorrect, then the AUV would have moved further across the flow in the direction away from the plume.

In any of these cases, the AUV should next maneuver relative to the most upflow detection point. This Reacquire maneuver must be achievable by the AUV and useful in any of the three circumstances.

The maneuver that we designed, referred to as a Bowtie, is depicted in Fig. 6. The Bowtie maneuver first tracks a line that starts on the side of the plume on which we estimate that the AUV is located. This line is angled –15 degrees relative to upflow. The upflow 15 degree angle is small enough so that the transition to Track-In is smooth, if chemical is detected. If that line completes without a detection, then the AUV transitions to the start of a second line that passes through the same center point, but has an angle of 15 degrees relative to upflow. In Fig. 6, the narrow lines indicate distances while the wide lines show the nominal AUV trajectory. If the Bowtie completes without a detection, then the last line would be followed by a clockwise turn toward downflow, which would have a radius of at least 5.0 m.

Therefore, this maneuver explores at least 13 m on each side of its center in the direction perpendicular to the flow.

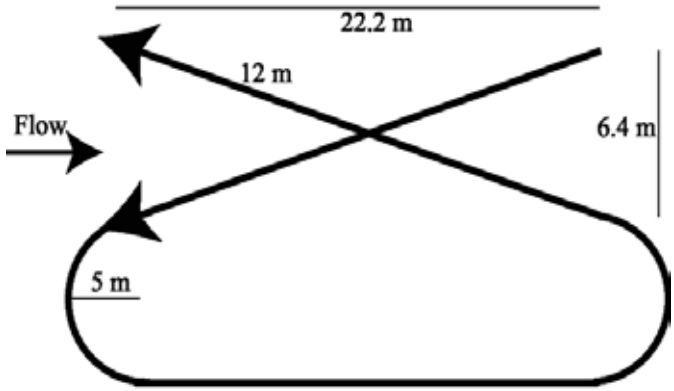


Fig. 6. Illustration of the BOWTIE maneuver used during the Reacquire maneuver. The image is not to scale.

The Reacquire behavior will perform at most N_re (>1) repetitions of the Bowtie in the vicinity of a single point on the lost point list. The first Bowtie is centered 10 m upflow from the most upflow point on the list of last detection points. The last Bowtie is centered on the most upflow point on the list of last detection points. The remaining (N_re-2) Bowtie centers are equally spaces between the first and last centers.

If this sequence of N_re Bowties completes without chemical detection, then the behavior removes the most upflow point from the list of last detection points. It then repeats the behavior at the most upflow point on the remaining list. This process repeats until a detection occurs or the list becomes empty. A detection at any time switches the behavior to Track-In. If the list becomes empty, then the AUV reverts to the Find behavior.

If the AUV started the Reacquire behavior upflow from the source, the shape of the Bowtie repetitions, as the center point moves downflow towards the last detection point, provides useful information for accurately declaring the source location. If the AUV starts the Reacquire behavior after crossing the plume without detecting, the repetitions of the Bowtie give the AUV several more chances to detect odor. If the AUV starts the Reacquire behavior across the flow from the plume, the repetitions of the Bowtie, at and upflow from the most upflow last detection point, will bring the AUV back towards the location where the plume is likely to be. The Bowtie is sufficiently wide so that it is able to recontact the plume as long as the plume has meandered across the flow less than 13 m away from the most upflow last detection point.

5.6 Declaration decision

The source declaration is not a separate behavior. Instead, it is a function that is called at the end of the Track-Out behavior. Each time that the Track-In ends, the last detection point is added to a list. That list is sorted according to distance along the direction of the flow. As long as the AUV is making progress up the plume, the first points on the list will be widely separated. When the AUV is near the source, the plume tracing maneuvers will cause several points on the list to be very near each other in the direction of the flow. When the first three points on the sorted list differ in the direction of the flow by less than 4 meters,

then the most upflow point on the list is declared as the source location. An additional error component is due to the fact that the vehicle navigation system may contain accumulated errors of approximately 10 m.
Note that the chemical source is on the bottom and that the AUV drives at a nonzero altitude above the bottom (altitude of 1.5 to 2.0 m is commanded). Therefore, the chemical plume does not rise to the altitude of the AUV, which is necessary for the AUV to detect the chemical, until the chemical has traveled some distance from the source in the direction of the flow. This distance is flow-dependent and is not known. Therefore, the declared source location is expected to have an error component, relative to the true source location, that is in the direction of the flow.

6. Field test results

Two variations of CPT algorithms were tested in four different sets of experiments. A first CPT algorithm, described with experimental results in (Farrell et al., 2003), was tested at San Clemente Island (SCI) CA in September 2002 and at SCI in November 2002. Based on the results of those tests, the Find, Reacquire, and Source Declaration behaviors were revised and the post-declaration maneuvers were added. The revised CPT strategy described herein using the parameters shown in Table 4 was experimentally tested at SCI in April 2003 and at Duck NC in June 2003 (Farrell et al., 2005). The April 2003 experiments successfully declared the source location on 7 of 8 experiments. The experiments included ground truth confirmation of declared source locations via sidescan sonar. The algorithms and field test results described herein, unless otherwise noted, are from the June 2003 experiments in Duck NC.

Symbol	Behavior	Value
λ	Find	5.0 s
β	Find	20 deg
L_u	Track_out	18.0 m
L_c	Track_out	18.0 m
N_re	Reacquire	2
K	Guidance	5.0
R	Guidance	10.0 m

Table 4. Parameter settings for CPT strategy for the April 2003 SCI and June 2003 Duck experiments.

Two types of missions were of interest during this set of experiments. The first mission type, labeled ST, contained a single chemical source in the OpArea. The ST mission was intended to find the plume, to trace a plume over a long distance, and to declare the source location. This mission demonstrates detection and tracing of plumes over long distances. The second mission type, labeled MT, may contain a few chemical sources in the OpArea. In an MT mission, the OpArea will be divided into subregions. The AUV will search each subregion for chemical until one of three events occurs. First, the search within a subregion may timeout. In this case, the subregions is declared source free and the AUV moves on to the next subregion. Second, the AUV may detect chemical and declare a source location within the region. It will then move on to the next subregion. Third, the AUV may trace chemical to the upflow edge of the region. In this case, a source will be declared at the intersection of the plume with the upper edge of the subregion and the AUV will move on to the next

subregion. When the declared source locations are analyzed at the end of an experiment it is up to the test director to decide whether source locations at the edge of a subregion are due to sources near that location or the result of plumes generated by sources in the adjacent region.

The AUV for these tests was the Albacore REMUS owned by SPAWAR in San Diego, CA. The REMUS was modified to contain a PC104 computer to run the AMP CPT algorithms. The AMP computer received sensor data from the REMUS computer via serial port, processed the sensor data, and output heading, speed, and depth/altitude commands to the REMUS computer via the same serial port. Up and down looking acoustic Doppler current profilers (ADCP) were onboard the REMUS. The AUV also had a CTD mounted onboard, but it was not used due to its slow response time. Also, the AUV used long baseline transducers with acoustic buoys in conjunction with dead-reckoning based on ADCP data to determine onboard AUV position. Finally, a fluorometer was mounted near the nose of the AUV. The fluorometer was capable of detecting Rhodamine dye from a source that was used to create the plume for these experiments. The fluorometer sample rate was 10 Hz.

Fig. 7 and Fig. 8 show the trajectory (solid line), chemical detection locations (x's), and declared source location (black dot) for two missions performed at Duck, NC in June of

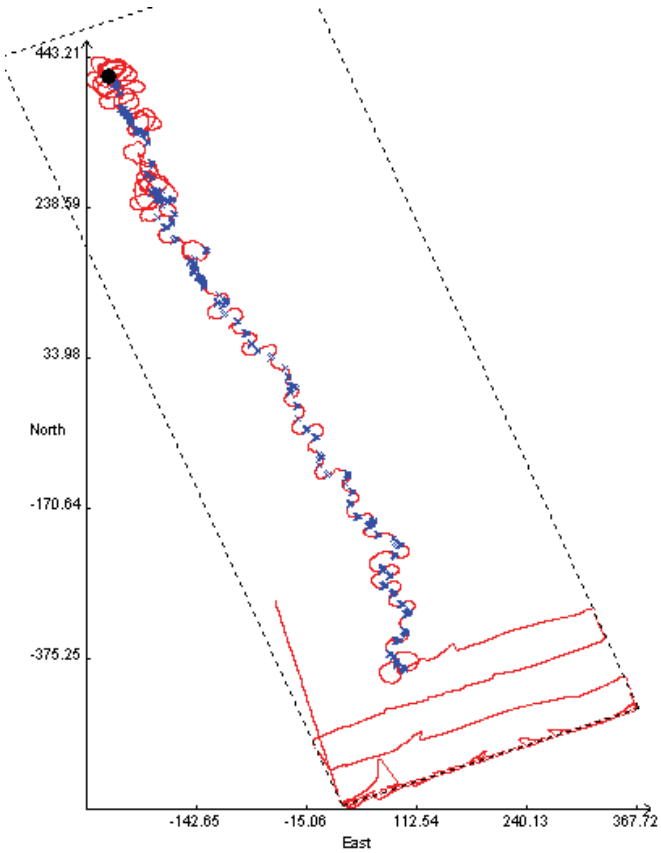


Fig. 7. Trajectory and chemical detection points. The dashed rectangle is the operating area boundary. The solid curve is the AUV trajectory. Each x marks the location of a chemical detection. The black dot at (N,E) = (414,-242)m marks the declared source location.

2003. The boundary of the OpArea is indicated by the dashed line. The figures use a coordinate system that is defined in the north and east directions relative to the center of the OpArea. These experiments were performed in 4-8 m of water. The bottom was gradually sloping from the coast. The coast is approximately 400 m to the left of boundary of the OpArea in both figures in this section. During all experiments included herein, the water column consisted of a top layer flowing northerly with a speed near 20-25 cm/s and a bottom layer flowing southerly with a speed near 10 cm/s. The depth of the boundary layer between these two flow regimes changed with location and time.

Fig. 7 shows the trajectory, chemical detection locations, and declared source location for an ST mission. For this mission, the OpArea was 367 x 1094 m (greater than 60 football fields). During this experiment, the flow calculated on the AMP varied in magnitude between 10 and 15 cm/s and in direction between 110 and 147 deg. For this experiment, the commanded speed was 2 m/s and the commanded altitude was 2 m. Note that the actual altitude varies by plus or minus 0.7 m relative to the commanded altitude. To challenge the CPT algorithm, we wanted the first chemical detection to occur as far as possible from the chemical source. Therefore, the source is located near the upflow edge of the OpArea and the AUV starts the mission near the downflow edge of the box. The AMP CPT algorithms start as soon as chemical is detected. This mission tracks the chemical plume for 976m between the first detection point and the declared source location. The source is declared at 36n11.028, 75w44.620. The ground truth source location is 36n11.035, 75w44.621 as found from sidescan data acquired during a post-declaration maneuver centered on the declared source location. The declared source location is 13 m south and 2 m east of the sidescan sonar location. Note that this error is predominantly in the direction of the flow, as expected.

Fig. 8 shows the trajectory, chemical detection locations, and declared source locations for an MT mission. The four subregions are outlined by dashed lines in Fig. 8. During this experiment, the flow calculated on the AMP varied in magnitude between 20 and 30 cm/s and in direction between 160 and 175 degree. For this experiment, the commanded speed was 2 m/s and the commanded altitude was 1.5 m. The southwest region is explored first. Chemical is detected and tracked for 351 m to the boundary between the southwest and northwest regions. The source for the first region is declared (correctly) at this boundary. Then, AMP drives the AUV to the northwest region. In the northwest region, the plume is tracked for an additional 180 m with a source declared at 36n11.034, 75w44.621. Sidescan sonar data confirmed the source at 36n11.037 , 75w44.622. The error between these locations is 6 m in the downflow direction. Note that this declared source is the same as that (for the same quadrant) from the missions shown in Fig. 7. Note that the latitude and longitude of the declared and sonar source locations match closely between these figures.

After declaring the source in the northwest region, AMP drove the AUV to the southeast region and restarted the CPT algorithm. During the transition from the northwest region to the southeast region using the Go To command, chemical detections are ignored. In the southwest region, chemical is detected and tracked a distance of 351 m to a source that is declared (correctly) on the boundary between the southeast and northeast regions. Then AMP drives the AUV to the northeast region. In the northeast region, the plume is tracked for an additional 185 m with the source declared at 36n11.079, 75w44.468. Sidescan sonar data confirmed the source at 36n11.087, 75w44.450. The error between these locations is 31 m in the crossflow direction. This crossflow error is clearly visible in the northeast region of Fig. 8. This crossflow error is an artifact of a navigation fix that occurred prior to the

declaration and the declaration logic that only accounted for position differences in the direction of the flow. This will be fixed in future versions of the algorithm.

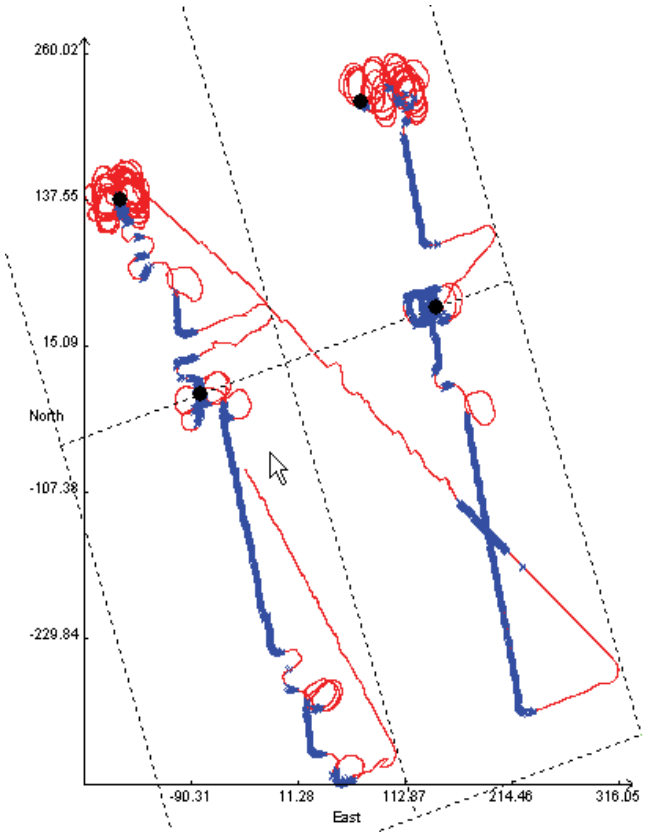


Fig. 8. Trajectory and chemical detection points. The dashed rectangle is the operating area. The solid curve is the AUV trajectory. Each x marks the location of a chemical detection. The black dots mark the declared source location.

Note that in spite of the exact same strategy and parameters being used in all runs, the nature of the trajectories shown in Fig. 7 and 8 during the plume tracing phase look different. Therefore, the differences in experimental conditions deserve comment. First, the mission shown in Fig. 7 was one of the first trials at Duck NC. Due to the fact that we were operating in an unknown environment, the commanded altitude for that mission was 2.0 m. For the mission corresponding to Fig. 8, the commanded altitude was 1.5 m. Analysis of the log files show that plume tracing for the mission of Fig. 7 frequently used the Track-Out behavior, which relies on large magnitude turns designed to cross the plume. Fig. 7 clearly shows this behavior. Plume tracing for the mission shown in Fig. 8 primarily used the Track-In behavior, since its small angle counterturning caused the AUV to drive up the main body of the plume. The difference in commanded altitudes could be the major reason for this difference, if the 2 m altitude of Fig. 7 only allowed the AUV to intermittently contact the top of the plume. Note also that in Fig. 8, as the AUV approaches the source, it must use the Track-Out behavior more frequently, because near the source the plume is still at a lower altitude.

The values of the parameters of the CPT strategy are summarized in Table 4. If β is increased, then the counterturns have a larger cross flow component. The tradeoff is that the larger crossflow component increases the probability that the AUV exits the plume from the expected edge (i.e., the variable LHS is more likely to be correct), but increases the length of the trajectory to get to the source. The variable λ should be larger than the intermittent chemical detection gaps while "in the plume;" however, the plume intermittency is dependent on characteristics of the flow and turbulence that are not known. Typical "in the plume" interpulse durations are less than 1s (Jones, 1983). As λ is increased, if chemical is not detected, then the distance that the AUV moves from the last detection point is increased. As long as this distance is less than L_u, then no backtracking is required. For these experiments, v_c = 2.0 m/s. Therefore, for λ = 5s, the distance traveled is 10 m which is less than L_u. The value of L_c was selected to ensure that, even with navigation errors (<10 m nominally) and with the Go To guidance command being satisfied when the AUV was within 10 m of the destination, the AUV would cross a line extending upflow from the last detection point. The value of N_re was set to 2. Increasing N_re causes the AUV to spend additional time searching upflow from each point on the lost_pnts list. This additional time is detrimental when the BowTie's are upflow from a false-alarm detection point. The values of K and R are dependent on the dynamic capabilities of the AUV. These values were determined in simulation and evaluated onboard the AUV prior to the CPT experiments described herein.

Note also, that the definition of a chemical detection implicitly contains two parameters: the detection threshold and the number of above threshold readings required to declare a detection. For all variations of CPT strategies that we performed during the three year program, the definition of a chemical detection was a concentration c(t)>4% of full scale (i.e., 0.2 V). This value was determined by analysis of chemical sensor data from the AUV operating in San Diego Bay (August 2002) in the absence of the chemical. In this scenario, the sensor readings were pure noise, but never surpassed 0.2 V. Therefore, we selected the threshold such that the probability of false alarm readings was extremely low. Therefore, any single sensor reading above threshold was registered as a chemical detection. The number of above threshold readings required to register a detection could be increased. This would decrease the probability of false alarms, but increase the probability of missed detections.

With the current AMP strategy and experimental results in mind, many alternative and possibly improved AMP strategies could be proposed. In fact, one of the goals of any experiment should be to identify areas for future improvements. Therefore, it is important to consider what lessons were learned in these experiments. First, care should be taken to ensure that the ADCP flow data corresponds to the flow layer containing the plume; however, this is not straightforward. For the Duck NC test location, the water is 4-8 m deep. The bottom boundary layer depth varied with time. The minimum safe AUV operating altitude was 1.5 m and the ADCP has an approximately 0.75 m deadzone prior to its measurement being accurate. Therefore, there were runs for which the upward looking ADCP was measuring the flow in the top layer instead of the bottom layer. Detecting and accommodating such events would require significant advancements for the planner and possible a conductivity, temperature, and depth (CTD) sensor with a fast response time. Second, some of the declared source locations had unexpected error in the crossflow

direction, which was unexpected. We believe that this error component is due to navigation fixes that occurred near the time of declaration and by the declaration logic that ignored separation in the crossflow direction. The source declaration logic described herein was based only on the along flow separation of points at which the plume was lost. The crossflow separation was ignored in the declaration process to decrease the time required to make a declaration. Accounting for crossflow separation in the declaration logic would improve the accuracy of the declaration and is straightforward to implement in the future. Third, the current AMP strategy used the chemical sensor in a Boolean mode even though the sensor did provide an analog reading. It is often suggested that the analog concentration could provide a useful indicator of the distance to the source; however, there are a few difficulties in this approach. First, the chemical source concentration would be unknown in a real application. Second, the rate of decay of the peak concentration reading as a function of the distance from the source is flow dependent and not known. Third, maximum sensed concentration along any transect is not necessarily the maximum concentration in the vicinity of that transect. Alternative, the analog sensor reading could have utility in experiments where multiple sources might generate overlapping plumes. In that scenario, a significant decrease in the maximum sensed chemical while moving upflow might indicate that a source has just been passed by while the AUV is still in the plume of another source. Such strategies were not required for this project.

It is also interesting to consider adaptation of the AMP strategy parameters based on distance from the source. For example, it might be more efficient to decrease L_u and L_c as the AUV gets nearer to the source. The difficulty in implementing such ideas is in evaluating the distance to the source when the source location is unknown. Early in the program, we hoped that the width of plume transects would be a useful indicator of the distance to the source. This proved futile for a variety of reasons: plume meander results in AUV transects being at different angles relative to the plume centerline; a variety of factors result in AUV transects being at different altitudes relative to the plume centerline altitude; and, the instantaneous plume width at a fixed distance from the source varies widely. Similarly, sensed chemical concentration is not a useful indicator of distance to the source since the source concentration is unknown and the sensed concentration at a fixed distance from the source varies widely.

7. Conclusion

This chapter has presented adaptive mission planning algorithms and experimental results for the first demonstration of chemical singal guided AUV. The experiments occurred in a near shore ocean environment. Plume tracing was demonstrated over distances of 975 m with average source declaration accuracy of approximately 13 m. This error includes the unknown distance required for the plume to rise to the altitude at which the vehicle is traveling.

The CPT planning algorithms were developed based on behavior based planning techniques that the CPT problem was divided into several simple sub-problems (e.g., find problem, tracing problem, reacquiring problem). The find problem is to search a potentially large area to detect the plume for the first time; the tracing problem is to trace the plume to its source once the vehicle detects the chemical concentration over a threshold; the reacquiring

problem initiates a local search based on knowledge of the flow and past detection information to reacquire contact with the plume if contact with the plume is lost. At last, we developed the coordinating methodologies to switch between these strategies in an intelligent manner.

8. References

Arbib, M. (1981). Perceptional Structures and Distributed Motor Control, In: *Handbook of Physiology - The Nervous Systems II,* Brooks (Ed.)

Arkin, R. & Murphy, R. (1990). Autonomous navigation in a manufacturing environment, *IEEE Trans. Robot. Automation,* Vol. 6, 445-454.

Basil, J. (1994). Lobster orientation in turbulent odor plumes: simultaneuos measuremetns of tracking behavior and temporal odor patterns. *Biological Bulletin*, Vol. 187, 272-273

Belanger, J. & Willis, M. (1998). Adaptive control of odor-guided location: Behavioral flexibility as an antidote to environmental unpredictability. *Adaptive Behavior*, Vol. 4, 217-253

Bell, W. & Tobin T. (1982). Chemo-orientation. *Biol. Rev.,* Vol. 57, 219-260

Berg, H. & A. Brown D. (1972). Chemotaxis in escherichia coli analysed by three-dimensional tracking. *Nature*, Vol. 239, 500-504

Berg, H. (1990). Bacterial microprocessing, In: *Cold Springs Harbor Symp. Quant. Biol.* 539-545

Berg, H. (1993). *Random Walks in Biology*. Princeton University Press

Berns, K.; Dillmann, R. & Hofsteller, R. (1991). An Application of a Backpropagation Network for the Control of a Tracking Behavior, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2426-2431

Braintenberg, V. (1984). Vehicles: Experiments in Synthetic Psychology, MIT Press

Brooks, R. (1986). A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation*, Vol. 2, Num. 1, 14-23

Cardé, R. & Mafra-Neto A. (1997). Insect Pheromone Research, In: Mechanisms of flight of Male moths to pheromone, pp. 275- 290. Chapman and Hall, New York

Dubins, L. (1957). On Curves of Minimal Length with Constraint on Average Curvature, and with Prescribed Initial and Termainal Positions and Tangents. *American Journal of Mathematics,* Vol. 79, 497-516

Dusenbery, D. (1992). *Sensory Ecology: How Organisms Acquire and Respond to Information*, W.H. Freeman, New York

Elkinton, J.; Cardé, R & Mason, C. (1984). Evaluation of time-average dospersion models for estimating pheromone concentration in a deciduous forest. *Journal of Chemical Ecology,* Vol. 10, 1081-1108

Farrell, J.; Li, W.; Pang, S. & Arrieta, R. (2003). Chemical Plume Tracing Experimental Results with a REMUS AUV. *MTS/IEEE Oceans 2003*

Farrell, J.; Pang, S. & Li, W. (2005). Chemical Plume Tracing via an Autonomous Underwater Vehicle. *IEEE Journal of Ocean Engineering,* Vol. 30, Num. 2, 428-442

Fossen, T. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley & Sons

Grasso, F.; Consi, T.; Mountain D. & Atema, J. (1996). Locating odor sources in turbulence with a lobster inspired robot, In: *From Animals to Animats 4: Proceedings of the Fourth*

International Conference on Simulation of Adaptive Behavior, P. Maes, M. J. Mataric, J.-A. Meyer, J. Pollack, and S. W. Wilson (Ed.), pp104-112, Cambridge, MA

Grasso, F (2001). Invertebrate-inspired sensory-motor systems and autonomous, olfactory-guided exploration. *Biological Bulletin*, Vol. 200, 160-168

Grasso, F.; Consi, T.; Mountain, D. & Atema, J. (2000). Biomimetic robot lobster performs chemo-orientation inturbulence using a pair of spatially separated sensors: Progress and challenges. *Robotics and Autonomous Systems*, Vol. 30, 115-131

Hassler, A. & Scholz, A. (1983) *Olfactory Imprinting and Homing in Salmon* Springer-Verlag, New York

Jones, C. (1983). On the structure of instantaneous plumes in the atmosphere. *Journal of Hazardous Materials*, Vol. 7, 87-112

Li, W. ; Ma, C. & Wahl, F. (1997). A Neuro-Fuzzy System Architecture for Behavior-Based Control of a Mobile Robot in Unknown Environments, *Fuzzy Sets and Systems*, Vol. 87, 133-140

Li, W.; Farrell, J. & R. T. Cardé (2001). Tracking of fluid-advected odor plumes: Strategies inspired by insect orientation to pheromone. *Adaptive Behavior*, Vol. 9, 143-170

Li, W.; Farrell, J.; Pang, S. & Arrieta, R. (2006). Moth behavior based subsumption architecture for chemical plume tracing on a remus autonomous underwater vehicle. *IEEE Transactions on Robotics and Automation*, Vol. 22, Num. 2, 292-307

Lohmann, K. (1992). How sea turtles navigate. *Scientific America*, Vol. 266, 82-88

Lytridis, C.; Kadar, E. & Virk, G. (2006). A systematic approach to the problem of odour source localisation. *Autonomous Robots*, Vol. 20, Num. 3, 261-276

Martinez, D. (2007). Mathematical physics: On the right scent. *Nature*, Vol. 445, 371-372

Murlis, J.; Elkinton, J. & Cardé, R (1992). Odor plumes and how insects use them. *Annual. Review of Entomology*, Vol. 37, 505-532

Mylne, K. (1992). Concentration fluctuation measurements in a plume dispersing in a stable surface layer. *Boundary-Layer Meteorology*, Vol. 60, 15-48

Naeem W.; Sutton R.; Ahmad S. & Burns, R. (2003) A review of guidance laws applicable to unmanned underwater vehicles. *The Journal of Navigation*, Vol. 56, Num 1, 15-29

Naeema1, W.; Suttona, R. & Chudley J. (2007). Chemical plume tracing and odour source localisation by autonomous vehicles. *Journal of Navigation*, Vol. 60, 173-190

Nevitt, G. (2000). Olfactory foraging by antartic proellariiform seabirds: Life at high Reynolds numbers. *Biol. Bull.*, Vol. 198, 245-253

Shraiman, B. & Siggia, D. (2000). Scalar turbulence. *Nature*, Vol. 405, No. 8, 639-646

Spears, D.; Zarzhitsky, D. & Thayer, D. (2005) Multi-robot chemical plume tracing. *Proceedingsof the 2005 International Workshop on Multi-Robot Systems*, pp. 211-222

Stacey, M.; Cowen, E.; Powell, T.; Dobbins, E.; Monismith, S. & Koseff J. (2000). Plume dispersion in a stratified, near-coastal flow: measurements and modeling, *Continental Shelf Research*, Vol. 20, 637--663

Sutton, O. (1947). The problem of diffusion in the lower atmosphere. *Quart. J. Roy. Meteorol. Soc.*, Vol. 73, 257-281

Sutton, O. (1953). *Micrometeorology*, McGraw-Hill, New York

Vergassola, M.; Villermaux, E. & Boris I. Shraiman (2007). Infotaxis as a strategy for searching without gradients. *Nature*, Vol. 445, 406-409

Vickers, N. (2000). Mechanisms of animal navigation in odor plumes. *Biological Bulletin*, Vol. 198, 203-212

Wiesburg, M. & Zimmer-Faust R. (1994). Odor plumes and how blue crabs use them in finding prey. *J. Exp. Biol*., Vol. 197, 349-375

Zarzhitsky, D.; Spears, D.; Thayer, D. & Spears W. (2004). Agent-based chemical plume tracing using fluid dynamics. *Lecture Notes in Arti¯cial Intelligence*, Vol. 3228 146-160

Zimmer, R. & Butman, C. (2000). Chemical signaling processes in the marine environment: *Biological Bulletin*, Vol. 198, 168—187

20

Development of Test-Bed AUV 'ISiMI' and Underwater Experiments on Free Running and Vision Guided Docking

Jin-Yeong Park[1], Bong-huan Jun[2], Pan-mook Lee[2] and Junho Oh[1]
[1]Humanoid Robot Research Center, KAIST
[2]Ocean Engineering Research Department, MOERI, KORDI
Republic of Korea

1. Introduction

In this chapter, development of a test-bed AUV is described. Free running test and vision guided docking are also presented.

Autonomous underwater vehicles (AUVs) have become a main tool for surveying below the sea in scientific, military, and commercial applications because of the significant improvement in their performance. Despite the considerable improvement in AUV performance, however, AUV technologies are still attractive to scientists and engineers as a challenging field. For example, multiple AUVs and underwater docking are recent challenging issues in the field of AUV technologies (Edwards et al., 2004; Fiorelli et al., 2004; Stokey et al., 2001; Singh et al., 2001). To successfully implement these new technologies in the field, a number of sub-functions have to be tested and verified in advance. They could be control, navigation and communication functions as well as basic functions for AUVs, including an emergency architecture for survival. Since it would be very expensive and time consuming to conduct all these tests at sea, researchers and engineers engaged in the operation and development of underwater vehicles need easier test schemes and faster feedback of results in an environment similar to that of the sea.

Underwater docking of an AUV to a launcher without surfacing allows the AUV have longer and more frequent investigations. Data uploading, mission downloading and recharge of batteries are essential duties of docking systems. Many research institutes have developed docking systems for AUVs. An ElectroMagnetic homing (EM) system was proposed as one of them (Feezor et al., 1997). A magnetic field generated by coils on the dock was used in that system. An AUV sensed this magnetic field and was guided to the dock. The range of the EM system was limited to 25-30m. An optical terminal guidance system was also introduced (Cowen et al., 1997). That system was simple but highly effective. The optical docking system provided targeting accuracy on the order of 1 cm under real-world conditions, even in turbid bay water. Autonomous docking demonstrations using an ultra-short baseline (USBL) acoustic homing array were shown in (Allen et al., 2006). The acoustic system was capable of acquiring a dock-mounted transponder at ranges of 3,000m or more.

The superiority of the vision system was described in Deltheil et al. (2000). Deltheil et al. (2000) introduced simulations of an optical guidance system for recovery of an unmanned underwater vehicle (UUV). Acoustic, magnetic and optical sensing methods were compared. It was shown that the optical method is superior and the best choice for the UUV recovery guidance system. Vision systems can be easy to operate, robust and fast analyzing (Deltheil et al., 2000). Lee et al. (2002) suggested a visual servo control algorithm using one camera and an optical flow equation. The optical flow equation was combined with the linearized equations of motion of an AUV. They derived a state equation for the visual servoing AUV. They supposed that a charge-coupled device (CCD) camera was installed at the nose of an AUV. Hong et al. (2003) introduced an algorithm for how to estimate relative pose and relative position between an AUV and an underwater dock. They supposed that lights were arranged on the circular rim of the entrance to the dock. Then, the lights were projected as an ellipse on the camera image plane. The geometric and perspective relationship of the circle in 3-dimension space and the ellipse in a projected 2-dimension plane were used in that scheme. From the geometric shape of the ellipse, relative distance and relative pose were estimated. One shortcoming of the above research is that, there were no underwater experiments to verify the algorithms in Deltheil et al., (2000) or Hong et al., (2003).

The Maritime and Ocean Engineering Research Institute (MOERI), a branch of the Korea Ocean Research and Development Institute (KORDI), has developed a model for an AUV named ISiMI. ISiMI comes from the name of a small but strong fish from an old traditional story in Korea. In English, ISiMI is the acronym for Integrated Submergible for Intelligent Mission Implementation. ISiMI is an AUV platform that satisfies the various needs of the experimental tests required for the development of control and navigational architectures or software algorithms for underwater flying vehicles. To carry out a number of tests while avoiding many difficulties in field trials, the first version of ISiMI was designed to cruise in the Ocean Engineering Basin (OEB), a facility in KORDI that simulates the ocean's environment.

A series of free running tests was conducted to investigate the dynamic characteristics of ISiMI and to validate its various functions as a test-bed AUV. These tests included a turning test, a zigzag test, an automatic heading test, a depth control test and a guidance control test. The experimental responses were analyzed and compared with the simulation responses to verify the discrepancies between the two sets of results.

This chapter presents a review of our research work on the development of the AUV ISiMI and its performance evaluation, by simulations and experimental tests. First, the design and implementation of ISiMI, including its positioning system in the OEB, will be presented. Second, a series of test results in the basin and a discussion of the results will be presented, with comparisons of the simulated and experimental outputs. Finally, the second generation ISiMI, ISiMI100, a sea-trial version of ISiMI, will be introduced.

This chapter also presents a final approach algorithm for underwater docking based on vision-guidance, as well as its experimental realization. Configuration of the vision-guidance system for ISiMI will be described. Next, the image processing method used to discriminate an underwater dock is explained. The arrangement of the lights on the dock and how to estimate the center of the dock from the position of the lights is described. The final approach algorithm guides the AUV from when it reaches a distance of 10-15m away of the dock until it reaches the dock. It generates reference yaw and reference pitch from the

estimated center and allows the AUV to track them. Underwater docking experiments are also introduced. The overall system validity was investigated and the final approach algorithm was verified. There was a position where the lights on the dock were outside of the camera viewing range when the AUV was close to the dock. It was expected that the vision-guidance would be invalid in this area. It was also expected that an algorithm based only on vision-guidance would be insufficient in the underwater experiments. Hence, an auxiliary method for precisely guiding the AUV in the area close to the dock is also suggested.

2. System design of the ISiMI AUV

2.1 Basic design concept

The basic design concept of the AUV ISiMI is hinged on the implementation of a small vehicle that can be easily launched, recovered and operated without special handling equipment. This concept of a small AUV is intended to provide researchers with fast experimental feedback on new algorithms and instruments necessary to develop AUV technologies. Several AUVs were reviewed to lead the concept of ISiMI. Sea Squirt AUV, which is 1m long and weighs 35 kg, was built in 1988 under the MIT Sea Grant to gather oceanographic data from the Charles River and serve as a test-bed for software and instrumentation, with component costs of $40,000. Then the MIT Sea Grant built Odyssey I and II for oceanographic research, with component costs of $50,000 and $75,000, respectively (Bellingham et al., 1993). Hydroid's REMUS (Allen et al., 1997; Stokey et al., 2001) and GAVIA AUV, developed by WHOI and Hafmynd, respectively, are small commercially available AUVs with diameters less than 25 cm.

MOERI-KORDI has a big square basin, OEB, for simulation of real sea conditions. OEB is 68.8m long, 37.2m wide and 4.5m deep. Wave, current and wind generators are installed in the basin. The purpose of ISiMI is to serve as a test-bed AUV for the development of AUV technologies with fast experimental feedback in an OEB environment. The size of ISiMI is constrained by the OEB environment, so ISiMI is able to run free in OEB. Its downsizing is at the highest level of the design process. Because the dimensions of the AUV conflict with the space and payload for instruments, however, the hull size and weight of ISiMI were determined in the spiral design process, which is a feedback design from the basic design to the detailed design.

In general, the hull shape of AUVs has mainly two types. The first is the cruising type, which looks like a torpedo, and which surveys from hundreds of meters to hundreds of kilometers. Its streamlined shape is adopted for AUVs to minimize the drag forces acting on the hull while the AUV is cruising. They use axial thrusters and control planes to control their motion. The other type of hull shape is the hovering type, which inspects a specific area from several meters to hundreds of meters. Several thrusters are installed to keep their precise positions and attitudes in hovering motion. Since ISiMI is a test-bed for underwater survey and docking technologies, the torpedo-type hull shape was adopted. Moreover, a pressure hull structure was chosen rather than an open frame structure for spatial efficiency.

2.2 Design goal

Considering ISiMI's basic design concept, we formulated its design goals as follows:

1. *Downsizing:* Its weight had to be 20–30 kg, including payloads for additional sensors or instruments. This scale can be handled by one or two people.

2. *Operating speed and duration:* The vehicle must have been able to operate for more than 3 h at a speed greater than two knots.
3. *Hull shape:* The hull had to have an experimentally known shape that would make it convenient to estimate the hydrodynamic coefficients of a mathematical model and that would minimize the drag force to increase the efficiency of the AUV's thrust. It also had to have space for additional instruments and to endure a depth rating of more than 20m.
4. *Computing and processing power:* The AUV's embedded computer system had to have enough computing power to process navigational and control algorithms and to gather data.
5. *Reliability and compatibility:* The system had to employ proven equipment to minimize the faults of subsystems and to increase the reliability of the entire system. The equipment had to be chosen from among the vendor-supplied instruments continuously and stably.
6. *Extendability*: The AUV had to have the capacity to have additional functions with minimal changes in the basic design of its mechanical and electrical systems.

3. System implementation of ISiMI AUV

3.1 Overview of ISiMI

In the consideration of the design concept, a first version of ISiMI was manufactured with fundamental sensors that were required to enable it to cruise in the OEB. The appearance of ISiMI is shown in Fig. 1. Adopting the modular design concept, we can insert additional mission sensors by extending the length of the AUV's mid-section hull. Fig. 2 shows the ISiMI hull that was extended by inserting an acoustic telemetry module, and Table 1 shows a comparison of the principal dimensions of ISiMI and the extended ISiMI. This chapter

Fig. 1. Appearance of ISiMI AUV

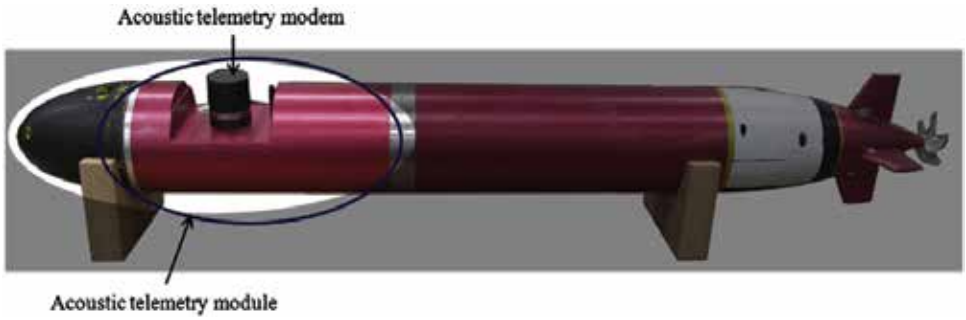


Fig. 2. Acoustic telemetry module

focuses on the first version of ISiMI AUV to cruise in the OEB. The ISiMI system is divided into a mechanical system, a control system and a communication system. Its mechanical

system includes its hull structure, its thruster and its control fins. Its control system includes a computer, electrical interface boards, sensors and software. Its communication system, which enables it to communicate with surface or other vehicles, includes wired and wireless LAN and a R/F modem. The extended ISiMI has an acoustic modem for underwater wireless communication. A new generation ISiMI100 will be equipped with a hybrid navigation system composed of an inertial measurement unit, a Doppler velocity log and range information (Lee et al., 2007; Lee and Jun, 2007).

Symbol	ISiMI	ISiMI100	Unit	Description
L	1196	1582	mm	Overall length
D	170	200	mm	Diameter
W	15	38	kg	Weight
U	1.0	1.5	m/s	Design velocity
u_{max}	2.0	2.0	m/s	Max velocity

Table 1. Dimensions of ISiMI

3.2 Mechanical system

The hull size of the ISiMI AUV is constrained by the space for the onboard instruments, and the hull shape is constrained by the AUV's hydrodynamic characteristics. The standard hull length and diameter are 1.2 and 0.17 m, respectively, considering the installation of the onboard equipment, which will be described in the following sections. The hull shape of the ISiMI AUV was designed based on the Myring hull profile equations (Myring, 1976), which are known as the best contours for minimizing the drag coefficient for a given ratio of body length and diameter. The dimensions of the ISiMI AUV are shown in Table 2. Its hull contour is shown in Figs. 3. Its mid-section is a pressure housing made of aluminum, and its nose section is a flooding hull made of polyurethane. Its tail-section tube, which includes a mounting jig for a BLDC motor and three linear actuators, is covered by a buoyant material.
The NACA 0012 cross-section was adopted for the control plane. The design parameters are listed in Table 3. Linear stepper motors were chosen for the fin actuators. The pair of rudders is controlled by a linear stepper motor, and the two stern elevators are independently driven by two linear stepper motors. Assuming the maximum torque required by the control fin is 10 kg cm, including the torque margin for mechanical loss, and assuming that the distance between the fin shaft and the driving axis is 5 cm, we chose a 25N stepper motor for each elevator and a 50N motor for the rudders. The capacity of the thruster motor was estimated based on the drag equation, described as

$$F_d = -\frac{1}{2} C_d \rho A_f u|u| \tag{1}$$

wherein F_d is the form drag acting on the hull, C_d is the drag coefficient, A_f is the maximum cross-sectional area of the hull, u is the advanced speed, and ρ is the water density. The required power capacity was obtained using:

$$P_M = -\frac{1}{2} C_d \rho A_f u|u| \cdot \eta \cdot u \tag{1}$$

wherein P_M is the estimated power capacity of the thrust motor, and η is the total efficiency, including the motor efficiency, the mechanical efficiency (including the friction loss), and the propeller efficiency. The calculation results are shown in Table 4 when the total efficiency was reduced to 0.4. To achieve more than the maximum design speed, an 80-watt BLDC motor with a reduction gear with a 5.8:1 gear ratio (made by Maxon Motors) was selected as the thruster motor. The selected propeller for the thrust system was KP452 S175, from the KORDI propeller series.

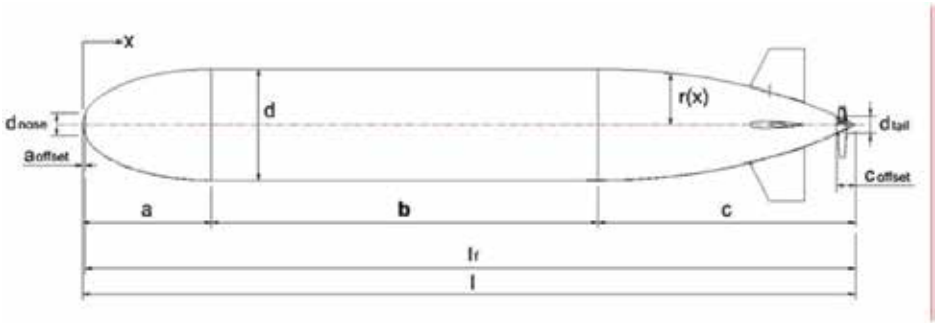


Fig. 3. Hull contour of ISiMI AUV.

Parameters	Value	Unit	Description	Remark
a	200	mm	Nose section	
b	600	mm	Mid-section	
c	400	mm	Tail section	
D	170	mm	Diameter	
a_{offset}	4	mm	Nose offset	
b_{offset}	30	mm	Tail offset	
l_f	1196	mm	Forward length	
l	1200	mm	Total length	

Table 2. Hull parameters of IsiMI

Parameters	Value	Unit	Description	Remark
S_{fin}	4819.5	mm^2	Planform area	
$X_{finpost}$	550	mm	Moment art wrt CB	
δ_{max}	20	degress	Maximum fin angle	

Table 3. Fin parameter of ISiMI

Speed	1m/s	1.5m/s	2m/s	2.5m/s
F_d(N)	2.2698	5.1071	9.0792	14.1867
P_d(W)	2.2698	7.6608	18.1584	35.4656
P_M(W)	5.6745	19.1514	45.3960	88.6641

Table 4. Drag force and required capacity of thrust motor for ISiMI AUV at Re = 1.6E+6, d = 0.17mm, C_d = 0.2, σ = 1.00E+03

3.3 Control System

The standard sensors of ISiMI include an AHRS (attitude heading reference system), a pressure sensor, a CCD camera, and a voltage sensor to check the battery voltage. The AHRS supplies information on the 3-axis angular velocities, 3-axis accelerations, 2-axis inclinations, and heading to the control system. The depth information is gathered from the pressure sensor. A CCD camera mounted on the nose is used to detect the underwater dock at the final stage of the underwater docking for the terminal guidance.

The general arrangement of the parts of ISiMI is shown in Fig. 6. The core of ISiMI's control system is a single-board computer interfaced with a frame grabber, a serial extension board, and a controller area network (CAN) module via a PC104 bus. Figure 4 shows a block diagram of ISiMI's control system. The operating system of the main computer is Windows XP, with real-time extension (RTX). The application software for the graphic user interface and the dynamic control of ISiMI is implemented with Visual C++. To interface sensors and actuators with the main controller, a sub-controller using a Micro Controller Unit (MCU) was developed. The sub-controller communicated with the main controller via a CAN. It controlled the linear actuators and digital and analog I/O intraface. A block diagram of the sub-controller is shown in Fig. 5. The operating duration of ISiMI was estimated at four hours with lithium-polymer batteries of a 207Wh capacity, as shown in Table 4. The total weight of ISiMI in air is 20 kg, including an additional payload of 5 kg. The CCD camera and the frame grabber constitued a vision-guidance system. Fig. 7 is a block diagram of the vision-guidance system. The CCD camera transmitted the Consultative Committee on International Radio (CCIR) signal to the frame grabber. The frame grabber was a PC/104+ type and grabbed image frames at 10-15 Hz. Specifications for each are shown in Table 5 and Table 6, respectively. The image frames were processed on a Windows timer which was not deterministic. Because the image processing could take a lot of time, it was not possible to do it using a deterministic timer. Results of the image processing were stored in the shared memory. The real-time controller extracted and used the results from the shared memory.

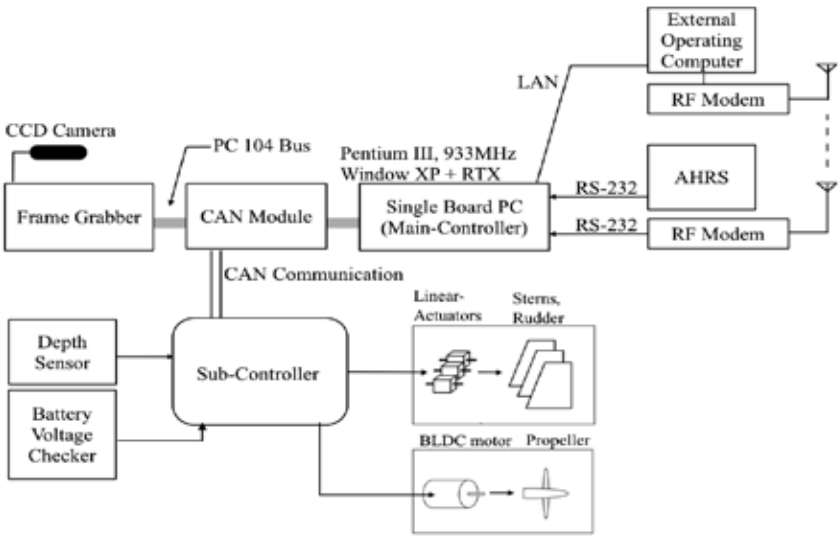


Fig. 4. Control system diagram of ISiMI.

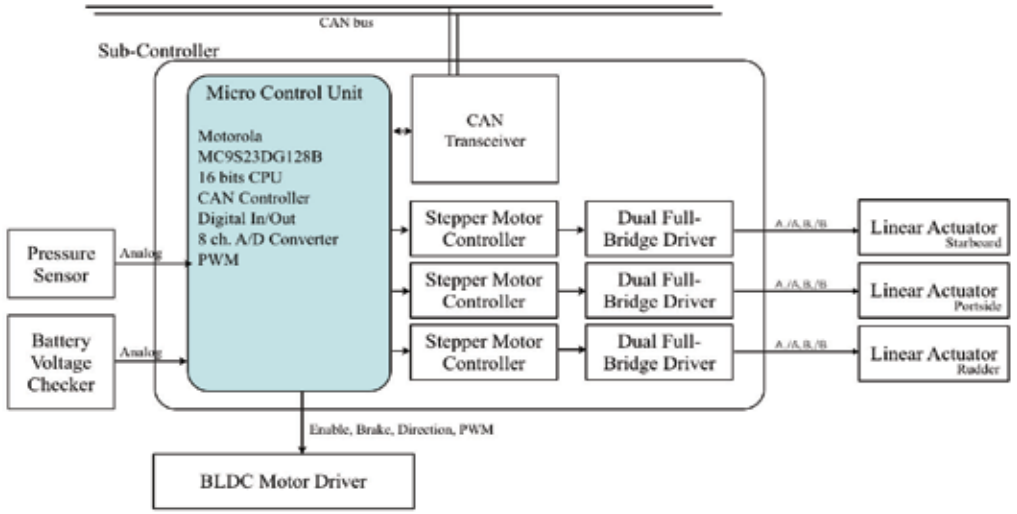


Fig. 5. Sub-controller.

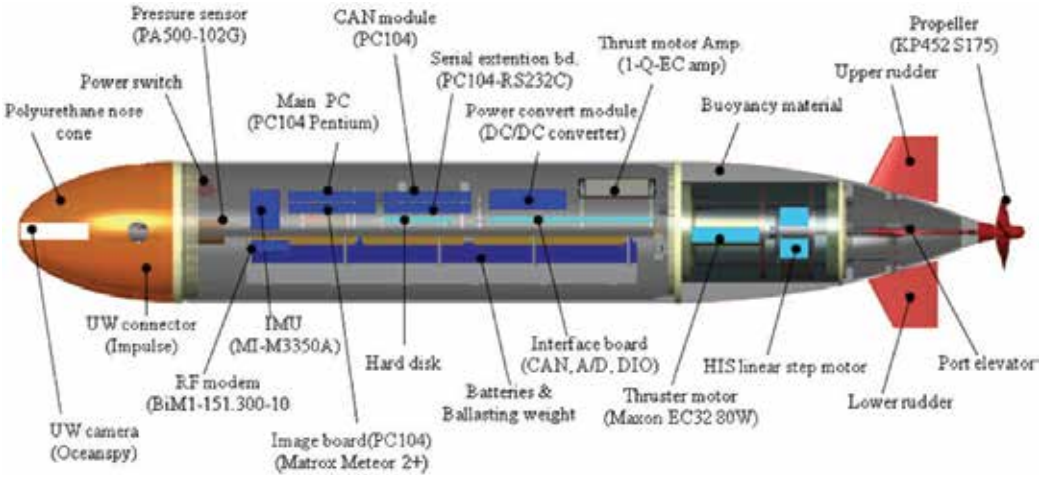


Fig. 6. General arrangement of the AUV ISiMI system.

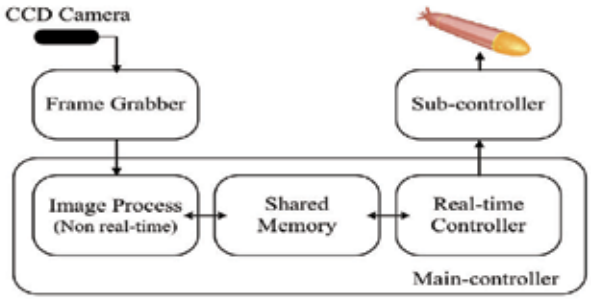


Fig. 7. Block diagram of the vision-guidance system.

Model	OceanSpy
Manufacturer	Tritech
Scanning	2:1 Interlace
Lens	3.6mm F2
Angular view in air	51^{o} vertical 40^{o} horizontal
Iris	Auto iris
Operating depth	6,000m water depth
Power	12-30V, 120mA
Dimension	10cm length, 3.4cm diameter

Table 5. Specifications of the CCD camera

Model	Matrox Meteor II+
Manufacturer	Matrox Imaging
Interface	PC/104+
Video source	NTSC, PAL, RS-170, CCIR
Channel	Up to 12 video inputs
Pixel format	RGB 8:8:8 or YU 4:2:2
Dimension	11.56cm length, 9.6cm width

Table 6. Specifications of the frame grabber

3.4 Communication system

A wireless local area network (LAN) was adopted as the communication system between ISiMI and a surface computer, which is used for offline communication for the mission allocation and data downloading. A wired LAN is additionally used as backup. A radio frequency (R/F) modem is installed for online communication between ISiMI and the surface computer for real-time data exchange. Generally, radio frequency cannot be used underwater because of the severe attenuation. We experimentally confirmed, however, that it is practical to transmit packets bi-directionally up to 3.5 meters deep in the OEB with an R/F modem with a bandwidth of 151.3 MHz and an output of 10 mW. With the R/F link, the user's commands and ISiMI's position are transmitted from the surface PC to ISiMI, and acknowledgments of reception are returned to the surface PC.

3.5 Localization in the OEB

The localization problem is a major challenge in underwater robotics. In the OEB environment, there is a severe constraint to using an acoustic positioning system due to the multi-path of the acoustic signal reflected from the wall and bottom of the OEB and from the free surface. Therefore, the image tracking system on the CPMC in the OEB is used for the non-contract position tracking of ISiMI. The CPMC has a three-degrees-of-freedom moving mechanism that enables ISiMI to follow the x-y position and heading of an object located

under the CCD camera. The position and heading of the object with respect to the CCD coordinates are transformed to the reference coordinates (the basin coordinates) using the position and heading of the camera with respect to the reference coordinates, which are calculated from the encoders of the CPMC actuators. The tracked position of ISiMI is transmitted to the ISiMI with a 2Hz bandwidth via a wireless R/F link between the surface computer and ISiMI in real time. The image tracking system installed under CPMC is shown in Fig. 8.

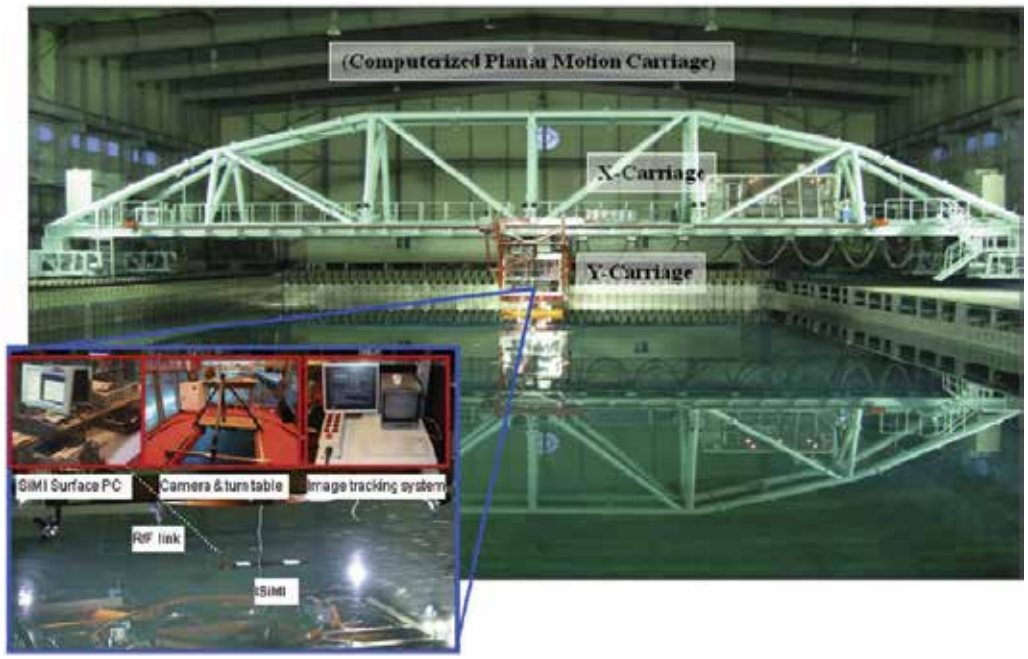


Fig. 8. CPMC and image tracking system of the OEB.

4. Numerical model and controller design

4.1 Numerical model

The three-dimensional non-linear dynamic equation of submersible motion has been described by Gertler and Hagen (1967), Feldman (1979) and Fossen (1994). Ignoring the effect of current, we can describe the dynamic equation of ISiMI as the form in the study of Feldman (1979) or Gertler and Hagen (1967). Considering the body coordinate system in Fig. 9 and classifying the terms in the dynamic equation, we get the following equation for ISiMI:

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{F}_{CC} + \mathbf{F}_{vh} + \mathbf{F}_{rest} + \mathbf{F}_{thrust} + \mathbf{F}_{fin} \tag{3}$$

wherein $\dot{\mathbf{v}} = \{u, v, w, p, q, r\}^{\mathrm{T}}$ is the linear and angular velocity vector with respect to the body coordinate frame, $\mathbf{M}$ is the inertial term including the added mass, $\mathbf{F}_{CC}$ is the coriolis and centrifugal force term for a rigid body, $\mathbf{F}_{vh}$ is the velocity-dependent hydrodynamic force acting on the body, $\mathbf{F}_{rest}$ is the restoring force, $\mathbf{F}_{thrust}$ is the thrust force, and $\mathbf{F}_{fin}$ is the lift and drag force on the fins. Each element term in (3) is listed in (Jun et al., 2008). The

hydrodynamic coefficients in the model were estimated using the Nernstein and Smith method and the Prestero method. The developed coefficients were non-dimensionalized with the length of the vehicle and are listed in (Jun et al., 2008). The roll damping coefficient was analogized from that of similar vehicles.

$$\dot{\boldsymbol{\eta}}_1 = \mathbf{J}_1(\boldsymbol{\eta}_2)\mathbf{v}_1 \;\Leftrightarrow\; \mathbf{v}_1 = \mathbf{J}_1^{-1}(\boldsymbol{\eta}_2)\dot{\boldsymbol{\eta}}_1 \tag{4}$$

$$\dot{\boldsymbol{\eta}}_1 = \mathbf{J}_1(\boldsymbol{\eta}_2)\mathbf{v}_1 \;\Leftrightarrow\; \mathbf{v}_1 = \mathbf{J}_1^{-1}(\boldsymbol{\eta}_2)\dot{\boldsymbol{\eta}}_1 \tag{5}$$

The velocity $\mathbf{v}$ in (3) can be written with respect to the earth-fixed coordinate frame with the transformation matrices $\mathbf{J}$ s as follows:

wherein $\mathbf{v}_1 = \{u, v, w\}^{\mathrm{T}}$ and $\mathbf{v}_2 = \{p, q, r\}^{\mathrm{T}}$ are the linear and angular velocities with respect to the body coordinate frame, respectively; $\boldsymbol{\eta}_1 = \{X, Y, Z\}^{\mathrm{T}}$ and $\boldsymbol{\eta}_2 = \{\phi, \theta, \psi\}^{\mathrm{T}}$ are the position and attitude vectors with respect to the earth-fixed frame, respectively; and $\mathbf{J}_1, \mathbf{J}_2$ are the linear and angular velocity transformation matrices referred to in (Jun et al., 2008). Based on the non-linear dynamics of ISiMI, we developed a simulation environment for it using MatLab and Simulink. All the simulation results presented in this chapter were derived from the simulation environment.

4.2 Controller design for free running tests

The sliding mode controller has been successfully applied to the control of underwater vehicles (Healey & Lienard, 1990; Utkin, 1997; and Lee, 1999). The sliding mode controller was used as the motion controller of ISiMI. The simplified equations for the steering and diving motions were derived by ignoring the cross-flow terms and higher-order terms. (Jun et al., 2008) The linearized steering system dynamics are given by:

$$\dot{\mathbf{x}}_h(\mathrm{t}) = \mathbf{A}_h\mathbf{x}_h(\mathrm{t}) + \mathbf{b}_h\delta_R(\mathrm{t}) \tag{6}$$

$$\mathbf{A}_h = \begin{bmatrix} I_z - \frac{\rho}{2}l^5 N_{\dot{r}} & 0 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \frac{\rho}{2}l^4 N_r & 0 \\ 1 & 0 \end{bmatrix}$$

$$\mathbf{b}_h = \begin{bmatrix} I_z - \frac{\rho}{2}l^5 N_{\dot{r}} & 0 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \frac{\rho}{2}l^3 U^2 N_{\delta_R} \\ 0 \end{bmatrix}$$

$$\mathbf{x}_h = \begin{bmatrix} r & \psi \end{bmatrix}^{\mathbf{T}}$$

The sliding poles of the steering system arbitrarily at [0 - 2], we got the steering control law as:

$$\delta_R = -0.0281r + 2.58\eta \tanh\left(r + 2\left(\psi - \psi_{com}\right)/1\right) \tag{7}$$

The linearized diving system dynamics are given by the third-order system:

$$\dot{\mathbf{x}}_v(\mathrm{t}) = \mathbf{A}_v \mathbf{x}_v(\mathrm{t}) + \mathbf{b}_v \delta_S(\mathrm{t}) \tag{8}$$

$$\mathbf{A}_v = \begin{bmatrix} I_y - \frac{\rho}{2} l^5 M_{\dot{q}} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \frac{\rho}{2} l^4 M_q & -z_G W & 0 \\ 1 & 0 & 0 \\ 0 & -u & 0 \end{bmatrix}$$

$$\mathbf{b}_v = \begin{bmatrix} I_y - \frac{\rho}{2} l^5 M_{\dot{q}} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \frac{\rho}{2} l^3 U^2 N_{\delta_S} \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{x}_v = \begin{bmatrix} q & \theta & z \end{bmatrix}^{\mathrm{T}}$$

Placing the sliding poles of the system at [0 - 2.5 - 2.4], we got the diving control law as:

$$\delta_S = 0.4414q + 0.5309\theta + 4\eta \tanh\left(q + 4\theta - 4\left(z - z_{com}\right)/0.65\right) \tag{9}$$

The line of sight (LOS) is the horizontal plane angle for the guidance function of ISiMI. It was derived from:

$$\psi_{com} = \tan^{-1}\left[\frac{\left(Y_k - Y(t)\right)}{\left(X_k - X(t)\right)}\right] \tag{10}$$

in which $\begin{bmatrix} X_k & Y_k \end{bmatrix}$ are the waypoints preprogrammed in the vehicle, and $\begin{bmatrix} X(\mathrm{t}) & Y(\mathrm{t}) \end{bmatrix}$ is the current location of the vehicle. The decision as to whether or not the waypoint has been reached is made from:

$$\rho^2(t) = \left[Y_k - Y(t)\right]^2 + \left[X_k - X(t)\right]^2 < \rho_0^2 \quad 0 < \lambda < 1 \tag{11}$$

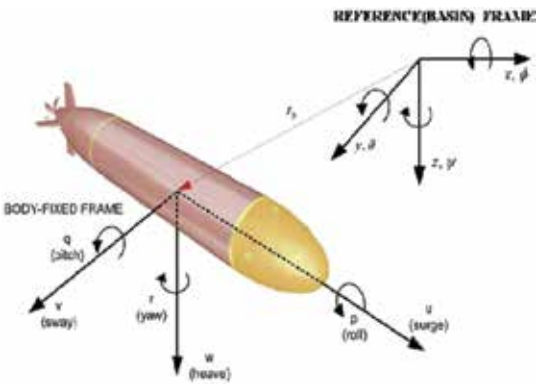


Fig. 9. Coordinate frame

5. Free running tests in the OEB

A series of free running tests of ISiMI was carried out in the OEB. The test was intended to determine ISiMI's maneuvering characteristics and to validate its basic functions as a test-bed AUV. The results might be used as references for the design of high-level control algorithms such as path planning or of the guidance system of ISiMI. ISiMI was easily launched without any special device, as shown in Fig. 10. The image tracking system of CPMC gathered the velocity and position of ISiMI during the free running test.

5.1 Advanced speed test

An advanced speed test was carried out, and the fulfillment of the design speed was verified. The advanced speed of ISiMI with respect to the propeller's rpm was determined, as shown in Fig. 11. The design speed (1 m/s) was reached at a propeller speed of about 1,000 rpm. By extrapolating the results shown in Fig. 11, it is estimated that the maximum speed (2 m/sec) was reached at 1,650 rpm.

Fig. 10. Launching and recovery of ISiMI in the OEB.

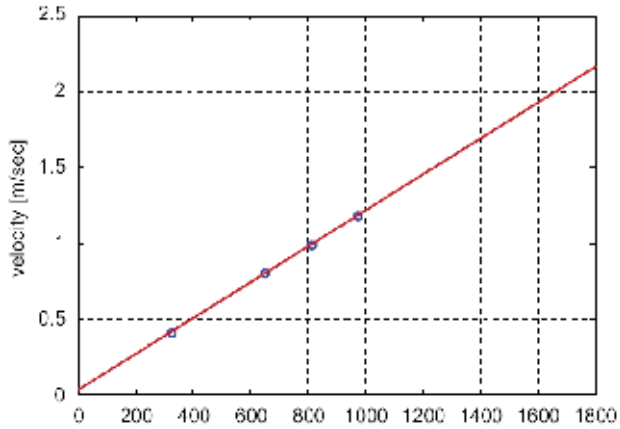


Fig. 11. Advanced speed test of ISiMI.

5.2 Open-loop zigzag test

The zigzag test was conducted to check the horizontal maneuvering property of ISiMI. The results of the test in the horizontal plane were plotted with the simulation results conducted

with the 6-DOF non-linear model in Fig. 12. The velocity of ISiMI was 0.8 m/sec and its rudder angle was toggled between -12.6 degrees and +12.6 degrees. The experimental results showed that the overshoot angle was in the range of 5~7.6 degrees. The hydrodynamic coefficients used in the simulation are listed in (Jun et al., 2008). The discrepancy between the simulation and experiment results shows that the numerical model has more yaw damping or less fin force than ISiMI. The experimental results will be used in system identification for the correction of the numerical model in future works.

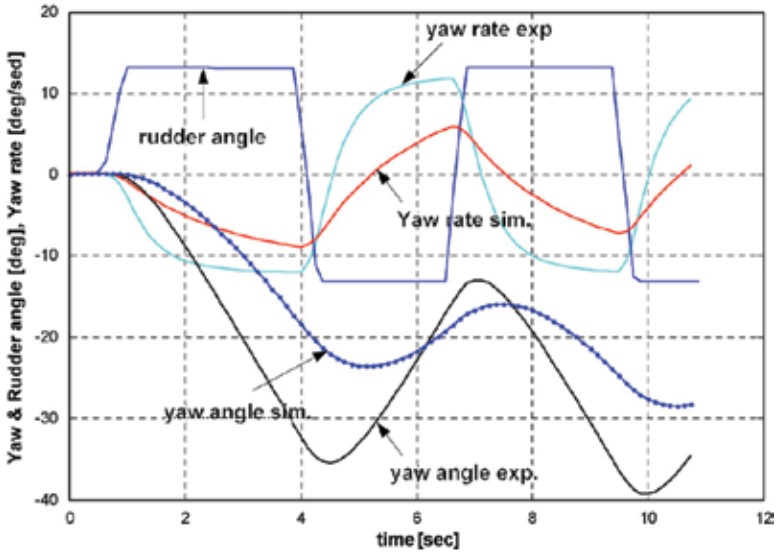


Fig. 12. Experimental results of open-loop zigzag test.

5.3 Open-loop turning test

To calculate ISiMI's turning radius, a turning test was carried out. The test results are plotted in Fig. 13, from which the turning characteristics of ISiMI were analyzed. The steady turning radius was about 6 m, and the steady turning speed and rate were about 0.6 m/sec and -6 degrees/sec when the rudder angle was 15 degrees at the advanced speed of 0.7 m/sec.

5.4 Closed-loop depth control

As the basic functions of an AUV test-bed, the depth control and waypoint tracking control functions were tested. A sliding mode controller was designed and tested for ISiMI's depth control. The experimental results are plotted with the simulation results of the linear and non-linear models in Fig. 14. The initial depth was 0.4 m, and the reference depth was 1 m. Both the experimental and simulated results showed good convergence with the reference depth. The settling time of the experiment results was longer, however, than that of the simulation results, and the amplitude of the pitch rate response in the simulation was larger than that in the experiment results. It is supposed that the discrepancies in the responses are due to the errors of the numerical and real model in pitch motion. A more exact numerical model will be estimated based on the experimental data using the system identification method in future works.

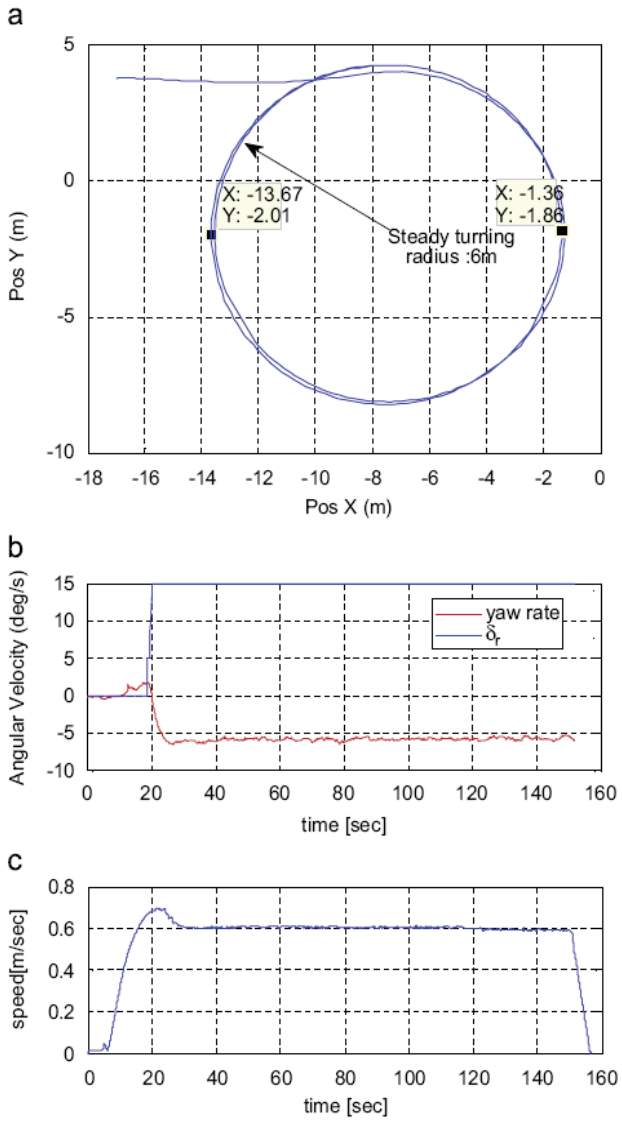


Fig. 13. Experimental results of open-loop turning test: (a) turning radius; (b) steady state turning rate and (c) steady state turning speed.

5.5 Waypoint tracking

ISiMI was guided to track a figure-eight trajectory to test the LOS algorithm described in Section 4.2. The position of ISiMI was measured using the image tracking system in CPMC, and transmitted to ISiMI via the R/F link. The position of the waypoints was pre-recorded in ISiMI's memory. The threshold level ρ_o in (11) was 1 meter. The results are plotted in Fig. 15. The yaw angle in Fig. 15 was controlled by the PD controller to follow the yaw reference generated by the LOS. It was verified that waypoint tracking, the basic function of the test-bed, was successfully achieved.

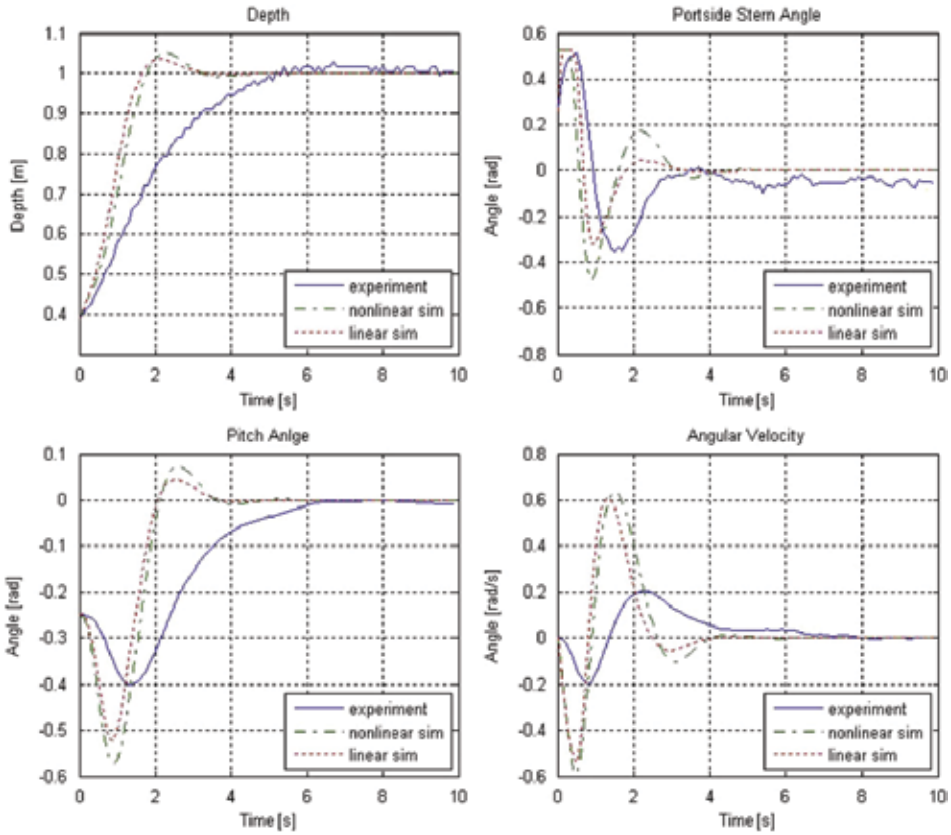


Fig. 14. Experimental results of close-loop depth control with sliding mode control.

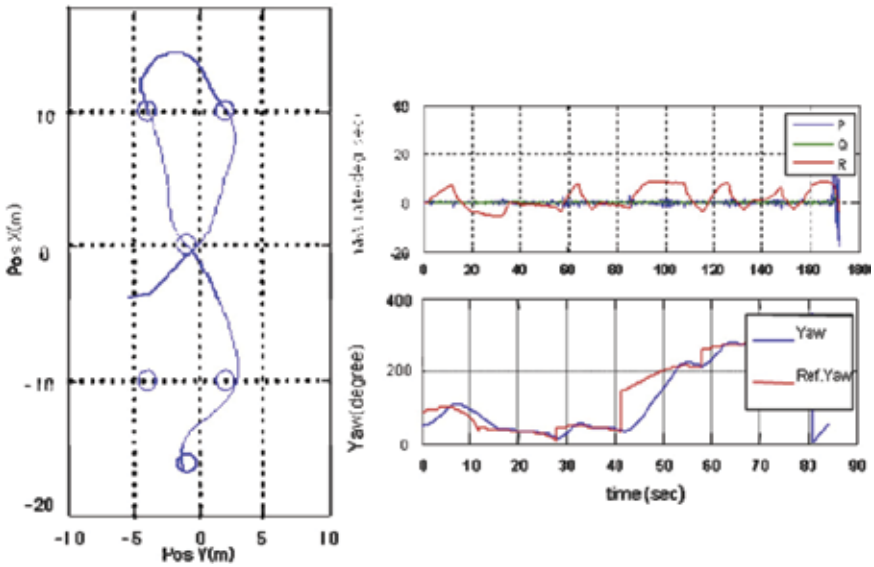


Fig. 15. Experimental results of waypoint tracking.

6. Image processing

In this chpater, image processing means detecting and discriminating the location of the lights on the dock and estimating relative position and distance between the AUV and the dock.

Figure. 16 shows a developed dock and the arrangement of the lights in the entrance of the dock. The diameter of the rim was 1m. Five lights were installed in the circular rim. The locations and brightness of the lights were adjustable. Before the image processing, it was necessary to adjust the intensity of the lights. If the lights is too strong, two or more lights may be misidentified as one light because of scattering. Proper intensity was determined through trial-and-error.

Figure 17 shows the coordinates of the vision system. The origin of the camera coordinate is at the center of the nose. The image coordinate is coincident with the camera coordinate except for the x-directional shift f (focus length).

The processing consists of 4 stages. (1) Image grabbing, (2) Binarization of the grabbed images, (3) Elimination of noisy luminaries and discrimination of the dock lights and (4) Estimation of the position and a distance of the dock center.

Stage (1) Raw images were grabbed. Lights were scattered, and this scattering made finding the exact positions of the light sources difficult.

Stage (2) To discriminate the lights installed around the dock entrance, the image processing unit classified each pixel of the raw images into two groups (a bright group and a dark group) using a pre-specified threshold value. Namely, the grabbed image was converted into a binary image. The lights of the dock were classified as the bright group, and the background became the dark group. In this process, salt-and-pepper noise was produced. Salt-and-pepper noise is the presence of single dark pixels in bright regions, or single bright pixels in dark regions. This is the natural result of creating a binary image via thresholding (Shapiro, 2001). This noise was removed using 3pixel by 3pixel masks. Appropriate size for the mask was determined by trial-and-error. After a pixel was classified in the bright group, the distance between that pixel and the nearest white pixel was investigated to identify each of the lights. If the distance was shorter than a pre-specified range, it was supposed that those two pixels belonged to one light region. If not, it was supposed that those two pixels belonged to two separate lights

Stage (3) Underwater, there were noisy luminaries that had to be eliminated. The luminaries are shown in Fig. 18. Some of the luminaries emitted light with an intensity similar to that of the dock lights. In such cases, ISiMI could be confused. Two particular problems were the presence of several lamps outside of the basin, and the fact that the dock lights were reflected back down from the surface of the water. Because these luminaries were interfered with the upper portion of the dock, they could be eliminated by processing the image frame from the lower-right pixel to the upper-left pixel. The processing was ended after five bright regions were acquired. All of the five lights had to be detected, i.e. the image processing required detection of all five lights. If one or more lights could not be detected, the image processing would fail. This was a defect of the developed processing. Stage (2) and Stage (3) were executed simultaneously. Due to Stage (2) and Stage (3), the dock lights were identified. Once the five dock lights were detected, a local searching area was generated and the image processor considered only this area in the next step.

Stage (4) From positions of the lights identified in the image coordinates, ISiMI estimated the position and the distance to the dock center. The center of the dock was calculated by

averaging the coordinates of the five lights. To estimate the distance, the number of white pixels in Light #5 was counted (Fig. 16). Because Light #5 was located relatively far from others, interference from scattering was relatively small. The distance information was used as one threshold to decide whether vision-guidance was valid or not.
If the coordinate of the n-th light is P_n then the center of the dock is given by

$$P_C(z) = \frac{\sum_{n=1}^{5} P_n(z)}{5}$$
$$P_C(y) = \frac{\sum_{n=1}^{5} P_n(y)}{5} \tag{12}$$

Fig. 19 shows the sequence of the introduced image processing. From the left, the raw image (left) and the binary image (center) are shown. There are some noisy luminaries that must be eliminated in the upper portion. The five white points in the lower portion are the dock lights. The last photograph (right) is a processed image. It shows elimination of the noisy luminaries and discrimination of the dock lights. The detected lights of the dock are marked by squares. The estimated center of the dock is also marked.

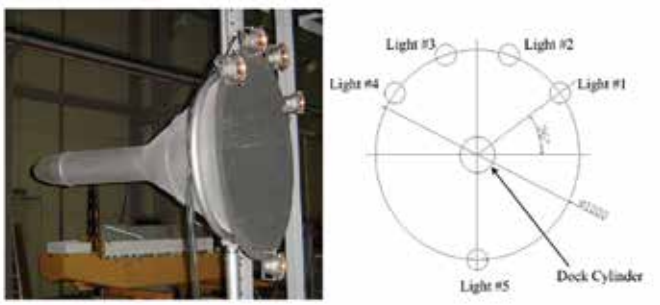


Fig. 16. The docking device with five lights around the rim of the entrance: The location and intensity of the lights is adjustable. (Left) Photo of the dock configuration and (Right) arrangement of the lights.

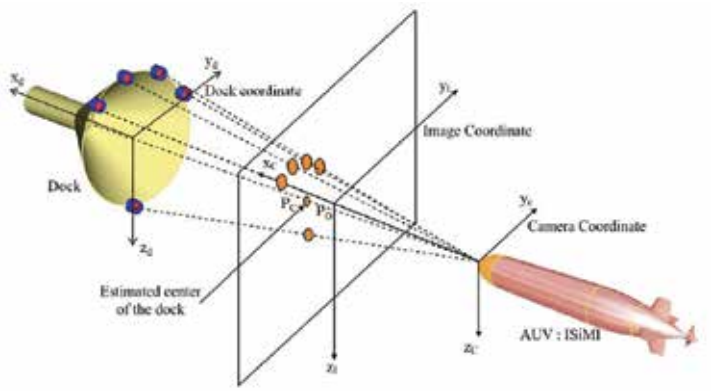


Fig. 17. Coordinates of the vision system.

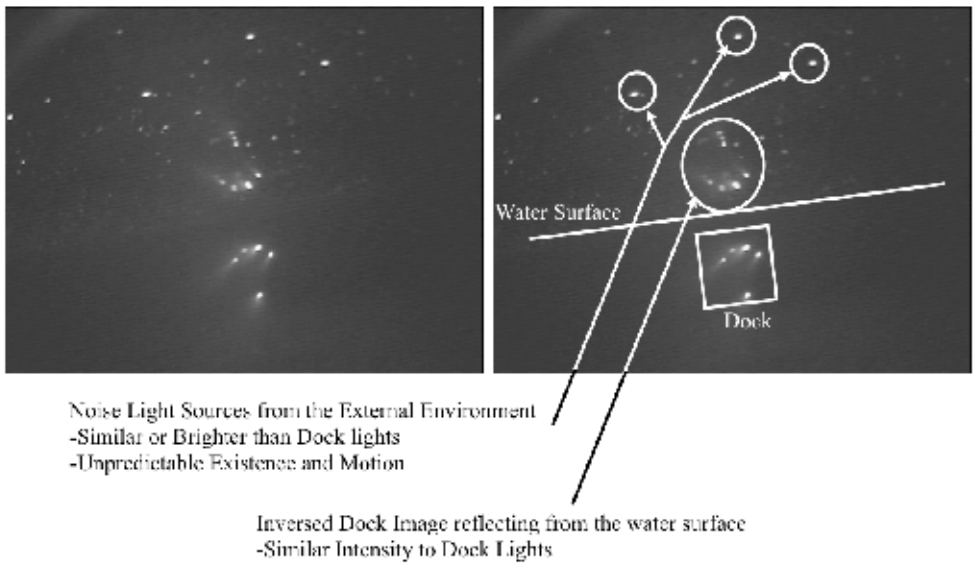


Fig. 18. Raw Image (left) and Noisy Luminaries (right): Several lamps were outside of the basin, and the dock lights are reflected down from the water surface.

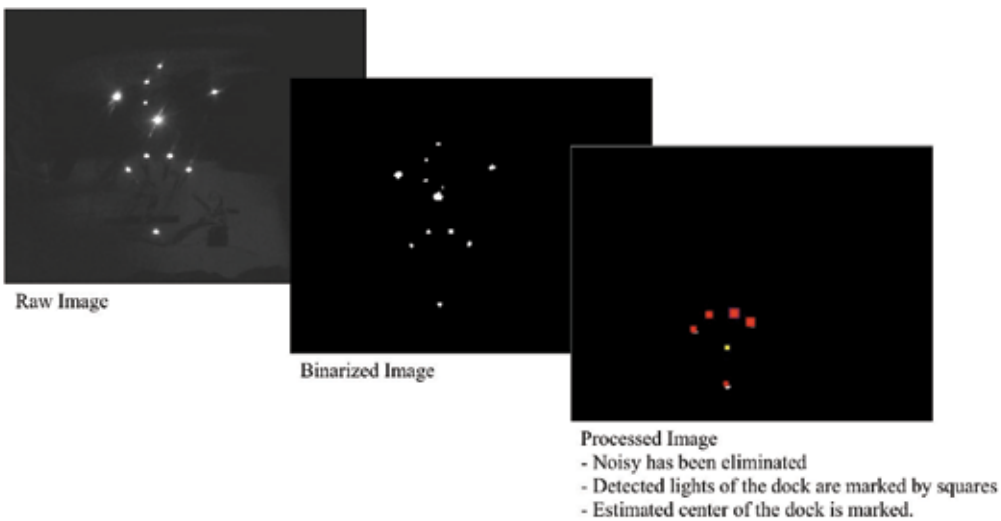


Fig. 19. Image process sequence (test screenshots was used): Raw image (left), binary image-some noisy luminaries remain. (center), and processed image - elimination of noisy luminaries and discrimination of the dock lights.

7. Final approach algorithm

It was first suggested by Deltheil et al (2000) that a vision system is suitable for docking because it offers simplicity, stealthiness and robustness. In this chapter, a final approach algorithm based on vision-guidance is suggested. It was supposed that the AUV could be

guided to the dock by controlling only yaw and pitch. This final approach algorithm generates reference yaw and reference pitch and makes the AUV track them.

The docking stage begins when the AUV arrives within 10-15 m in front of the dock. The docking stage of the return process is subdivided here into two stages because there exists an area where the dock lights are out of the camera viewing range when the AUV is close to the dock. Figure 20 shows the first and second stages. During the second stage the AUV is about 1.4m from the dock, and the lights of the dock are out of the range of the camera. The essential difference of the second stage is the manner of generating reference yaw for steering motion and reference pitch for diving motion. During both parts of the docking stage, a conventional Proportional-Derivative (PD) control is applied to track the references. Values of these gains were tuned by trial-and-error using the results of the simulations and underwater experiments.

A. The first stage

In this stage, reference yaw and pitch were generated based on vision-guidance. All dock lights were located in the viewing range of the CCD camera. This vision-guidance controller generated reference yaw and reference pitch using the estimated center of the dock. A discrepancy between the estimated dock center and the origin of the image coordinates became an error input of the vision-guidance controller. Fig. 21 is a block diagram of the vision-guidance control. A Proportional-Integral (PI) controller was used to generate reference yaw and pitch from the position error. To eliminate steady-state error, I-control was used. By conducting repeated underwater experiments, values of the PI gains were tuned.

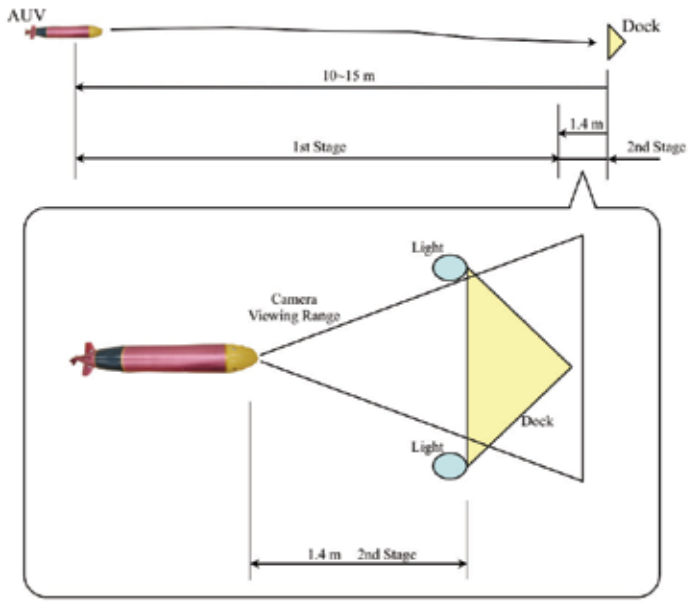


Fig. 20. The 1st stage and the 2nd stage of docking approach

B. The second stage

When the distance estimated by the image processing became smaller than a pre-specified threshold value, the second stage began. In this area, the last reference yaw and pitch

become fixed. Because the AUV is very close to the dock, it was supposed that changing yaw or pitch could be dangerous and keeping the final references would be plausible. This method is referred to as 'attitude keeping control.' (Park et al., 2007) During this phase, ISiMI becomes blind and simply tracks these final fixed references until contacting the dock. Fig. 22 shows a flow chart of the final approach algorithm.

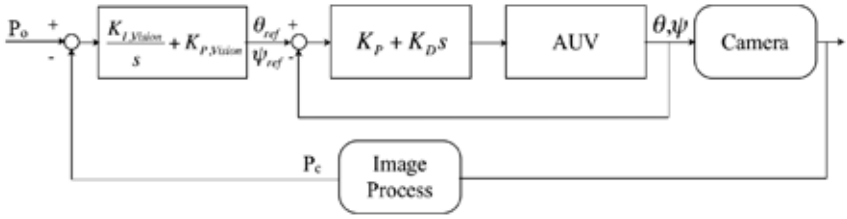


Fig. 21. The vision-guidance control algorithm. P_o is the origin of the image coordinate frame. P_c is the estimated center of the dock. θ is pitch, ψ is yaw. θ_{ref} and ψ_{ref} are generated reference pitch and yaw, respectively.

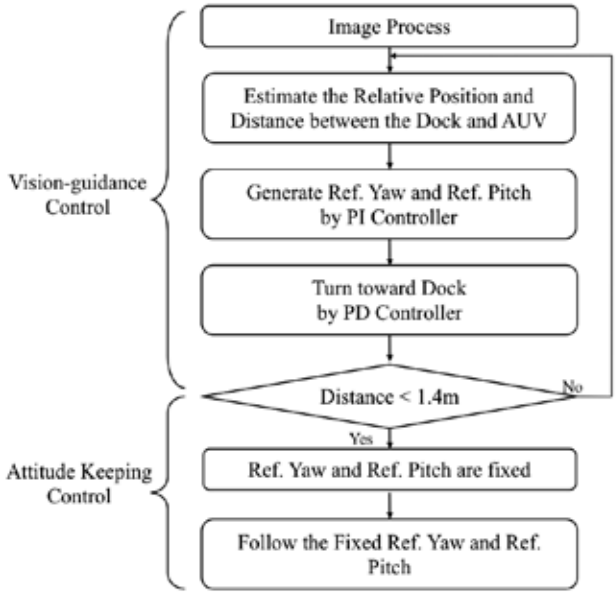


Fig. 22. Flow chart of the final approach algorithm.

8. Underwater docking experiments

The goal of the experiments was to verify the final approach algorithm and system validity. Figure 23 describes the initial start point for the final docking approach. It shows a top view (left) and a side view (right) of the initial start conditions. The dock was placed within viewing range of the camera. The center of the dock was placed at a depth of 1.5m. The dock was introduced by (Lee et al, 2003), (Park et al, 2007). The dock was funnel-shaped. This shape makes it possible for the AUV to dock successfully through sliding even if she approaches obliquely. The dock used an external power source.

Because robustness against disturbance has not yet been developed and this attempt was during the early stages of development, some restrictions were applied. There was no current and there were no waves. The dock was fixed on the basin floor. The water was clean. ISiMI was operated using a wired LAN communication. RF wireless communication was not suitable to receive the large amount of image data necessary. The wireless LAN was disconnected when the AUV submerged. The R.P.M. of the thrust propeller was invariant and the forward speed was about 1.0m/s. The relation between R.P.M and speed was determined by (Jun et al. 2008). There was no speed control. Experiments without the attitude keeping control and experiments with the attitude keeping control were conducted separately.

A. Underwater docking experiment without the attitude keeping control

Only the vision-guidance control was applied. No distance estimation was applied. ISiMI depended on the camera until contact with the dock. In Fig. 24, pixel errors are plotted against time. A pixel error is defined as deviation between the origin and the estimated center of the dock center in the image coordinate. The pixel errors decreased and were regulated during the first 9 seconds of the test. However, between seconds 9-15, there were discontinuous oscillations. These oscillations were caused by the defect of the image processing system to process, not by actual motions of the AUV, i.e. one more light moved out of the camera viewing range. The AUV became confused and it could not find the center of the dock. This occurred when the AUV was in the second stage area. To estimate the center precisely, all five lights were required, but in this area, the AUV could not see all of them. It was found that the AUV had some head-on collisions with Light #5 or the inner plane of the dock. She performed imprecise final approaches and suffered collisions with the dock. Fig. 25 is a sequence of continuously grabbed images taken by an underwater camera. (a) ISiMI starts, (b) she cruises to (c) the dock, (d) an imprecise approach near the dock, (e) after a collision, she rebounded and (f) she could not enter the dock. Thus, it was proven that the vision-guidance control was not unnecessary during this part of the docking procedure.

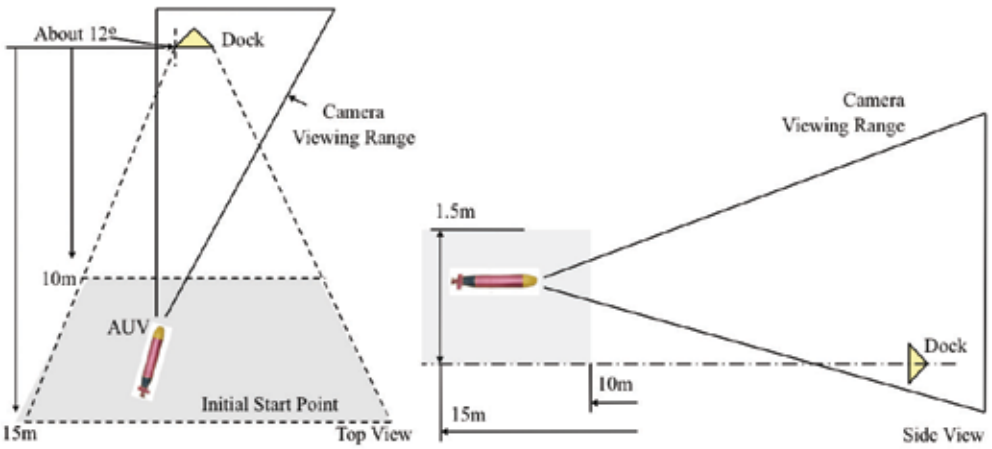


Fig. 23. Initial start point: (left) top view and (right) side view

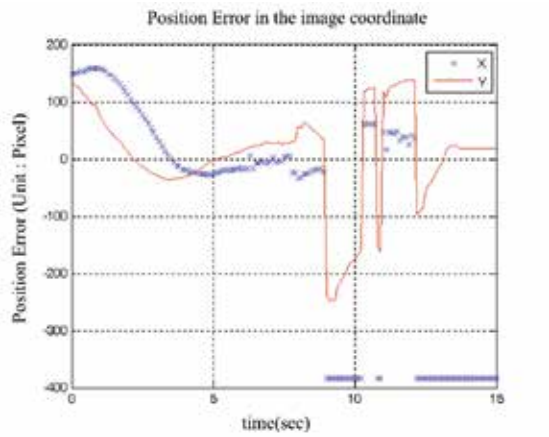


Fig. 24. Position error (unit: pixel) in the image coordinate. The vision-guidance control was applied through all intervals. (1) t = 0-9seconds : Errors are decreasing. In this interval, all 5 lights were in the viewing range of the camera. The AUV was able to estimate the center precisely. (2)t = 9-15seconds : one or more lights were out of the viewing range of the camera. Precise estimation of the center became impossible. The oscillation was caused by the defect of information from image processing rather than actual motion of the AUV.

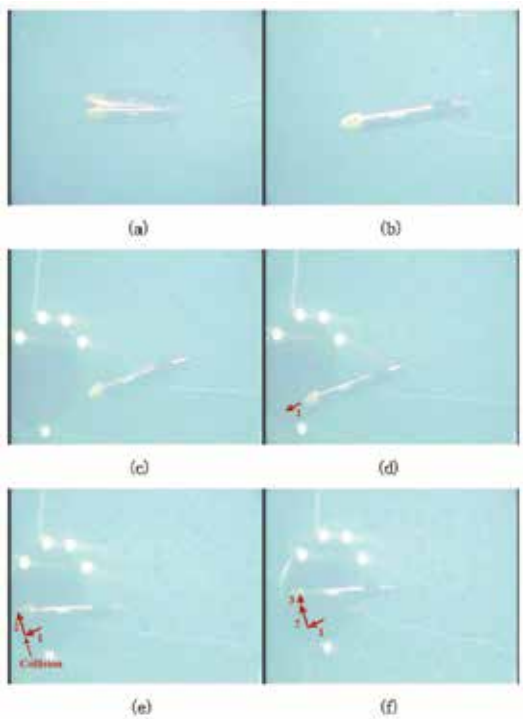


Fig. 25. Docking : Grabbed images by an underwater camera (Arrows indicate moving directions of the AUV): (a): ISiMI starts, (b): She cruises to (c) the dock, (d): An imprecise approach near the dock, (e): After a collision, she rebounded, and (f): She could not enter the dock.

B. Underwater docking experiment with the attitude keeping control
The attitude keeping controller was applied when ISiMI was near the dock. Image processing was used to estimate both the location of the center and the distance to the dock. The patterns were similar to that of Fig. 24 during the first 9 seconds of the test. Oscillations of the sort encountered during the first test were anticipated after 9 seconds. However, after the vision-guidance control was stopped, the reference yaw and pitch were fixed by the attitude keeping controller. In Fig. 26, the solid lines are the yaw(the upper graph) and pitch(the lower graph) measured by AHRS. The short-dash lines are the generated reference yaw and pitch. After 9 seconds, the references were fixed. The long-dash lines show the fixed references and the AUV tracked them. Fig. 27 shows the moment of docking. The

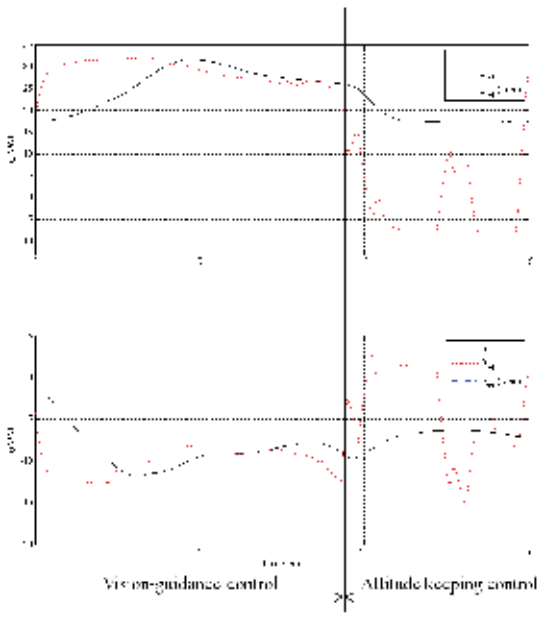


Fig. 26. Final approach: (upper) Yaw, Ref. yaw and fixed ref. yaw (lower) Pitch, Ref. pitch and fixed ref. pitch are shown respectively. After 9 seconds, ref. yaw and ref. pitch were fixed. (1)t = 0-9seconds : The vision-guidance control was applied. In this interval, all 5 lights were in the viewing range of the camera. (2)t = 9-15seconds : At t = 9.4seconds, the attitude keeping control began and the references were fixed.

Fig. 27. Docking: (left) The original photograph. The original photograph was sharpened anad the edge of ISiMI was emphasized to make her more easy to recognize. The white arrow indicates ISiMI (right).

original photo was sharpened and the edge of IsiMI was emphasized in order to make her more easy to recognize. The photo shows that ISiMI was going into the dock with a more precise approach.

9. Conclusion

In this chpater, the design, implementation and test results of a small AUV named ISiMI are presented. The AUV, ISiMI, developed in KORDI is a test-bed for the validation of the algorithms and instruments of the AUV. For fast experimental feedback on new algorithms, ISiMI was designed to be able to cruise in the Ocean Engineering Basin environment at KORDI. The zigzag test and the turning test were carried out to check ISiMI's maneuvering properties. The depth control and waypoint tracking tests were carried out to validate the feedback controller of ISiMI. The experiment results were compared with those of the simulation. The research works were fed back to the design and implementation of a 100m-class AUV named ISiMI100. ISiMI100 is equipped with additional sensors such as a doppler velocity log, an acoustic telemetry modem, an obstacle avoidance sonar, a range sonar, and a GPS module. A photo of ISiMI100 is shown in Fig. 28. The mission test of ISiMI and the sea trial of ISiMI100 remain to be performed in future works.

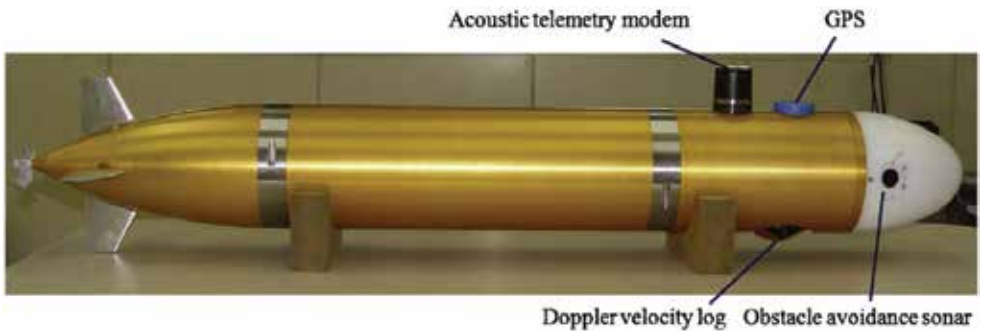


Figure 28. Sea-trial version of ISiMI AUV named ISiMI100

A final approach algorithm based on vision guidance for the underwater docking of an AUV was developed and introduced. The algorithm allowed the tested AUV to identify dock lights, eliminate interfering luminary noises and successfully estimate both the center of the dock and the distance to it during the first stage of the docking sequence despite the fact that the AUV was unable to detect the dock lights when close to the dock. The final approach algorithm based on vision guidance did guide the AUV to the dock successfully. The area where the lights were out of the camera viewing range occasioned confusion, as expected, but the attitude keeping control was able to keep the AUV on the way to the dock. Underwater docking experiments showed the necessity of the attitude keeping control. The use of the attitude keeping control as well as the vision-guidance control improved the precision of docking performance. The fixed references guided the AUV more precisely and safely. Although the docking experiments were conducted under controlled conditions, the results of the experiments showed the utility and potential of the vision-based guidance algorithm for docking.

Future problems include successfully docking when (1) the dock is moving, (2) the dock is placed out of the camera viewing range at the beginning of a return process, and (3) currents

and waves are present. Generation of the optimized path from any initial start point to the dock is also a subject for future study.

10. Acknowledgments

This work was supported in part by MLTMA of Korea for the "development of a deep-sea unmanned underwater vehicle," and KORDI, for the "development of ubiquitous-based key technologies for the smart operation of maritime exploration fleets."

11. References

Allen, B., Stokey, R., Austin, T., Forrester, N., Goldsborough, R., Purcell, M., von Alt, C., 1997. REMUS: A small, low cost AUV; system description, field trials and performance results. Proceedings of the Oceans Conference pp 994-1000.

Allen, B., Austin, T., Forrester, N. Goldsborough, R. Kukulya, A., Packard, G., Purcell. M., Stokey, R., 2006. Autonomous docking demonstrations with enhanced REMUS technology, Proceedings of OCEANS 2006. MTS/IEEE, pp. 1-6.

Bellingham, J.G., Bales, J.W., Goudey, C.A., Consi, T.R., 1993. Performance characteristics of the Odyssey AUV. Proceedings of the Eighth International Symposium on Unmanned Untethered Submersible Technology (AUV '93), pp. 37-59.

Cowen, S., Briest, S., Dombrowski, J., 1997. Underwater docking of autonomous undersea vehicles using optical terminal guidance, Proceedings of OCEANS '97. MTS/IEEE, Vol. 2, pp. 1143-1147.

Deltheil, C., Didier, L., Hospital, E., Brutzman, D. P., 2000. Simulating an optical guidance system for the recovery of an unmanned underwater vehicle, IEEE Journal of Oceanic Engineering, Vol. 25, No. 4, pp. 568-574.

Edwards, D.B., Bean, T.A., Odell, D.L., Anderson, M.J., 2004. A leader-follower algorithm for multiple AUV formations. Proceedings of IEEE/OES. pp 40-46.

Feezor, M. D., Blankinship, P. R., Bellingham, J. G., Sorrell, F. Y., 1997. Autonomous underwater vehicle homing/docking via electromagnetic guidance, Proceedings of OCEANS '97. MTS/IEEE, Vol. 2, pp. 1137-1142.

Feldman, J., 1979. DTNSRDC Revised Standard Submarine Equations of Motion. DTNSRDC/SPD-0393-09.

Fiorelli, E., Leonard, N.E., Bhatta, P., Paley, D., Bachmayer, R., Fratantoni, D.M., 2004. Multi-AUV control and adaptive sampling in Moterey Bay. Workshop on Multiple AUV Operations (AUV04). pp 134-147

Fossen, T.I., 1994. Guidance and Control of Ocean Vehicles. John Wiley & Sons Ltd.

Gertler, M., Hagen, G. R., 1967. Standard equations of motion for submarine simulations, NSRDC Report No. 2510.

Healey, A.J., Lienard, D., 1993. Multivariable sliding-mode control for autonomous diving and steering of unmanned underwater vehicles. IEEE Journal of Oceanic Engineering, 18 (3), 327-339.

Hobson, B. W., McEwen, R. S., Erickson, J., Hoover. T., McBride, L., Shane, F., Bellingham, J. G., 2007. The development and ocean testing of an AUV docking station for a 21" AUV, Proceedings of OCEANS 2007.

Hong, Y. H., Kim, J. Y., Lee, P. M., Jeon, B. H., Oh, K. H., Oh, J. H., 2003. Development of the homing and docking algorithm for AUV, Proceedings of the Thirteenth International Offshore and Polar Engineering Conference, pp. 205-212.

Jeon, B.H., Lee, P.M., Li, J.H., Hong, S.W., Kim, Y.G., Lee, J., 2003. Multivariable optimal control of an autonomous underwater vehicle for steering and diving control in variable speed, Proceedings of Oceans Conference, San Diego, pp 2659-2664.

Jun, B. H., Park, J. Y., Lee, P. M., Ma, S. J., Kim, Y. S., Oh, J. H., Lim, Y. K., 2007. Design, implementation and free running test of ISiMI; an AUV for cruising in ocean engineering basin environment, Proceedings of OCEANS 2007 IEEE Aberdeen.

Jun, B. H., Park, J. Y., Lee, F. Y., Lee, P. M., Lee, C. M., Kim, K. H., Lim, Y. K., Oh, J. H., 2008. Development of the AUV 'ISiMI' and a free running test in an Ocean Engineering Basin, Ocean Engineering, http://dx.doi.org/10.1016/j.oceaneng.2008.07.009

Kim, J. Y., Park, I. W., Oh. J. H., 2006. Experimental realization of dynamic walking of the biped humanoid robot KHR-2 using zero moment point feedback and inertial measurement, Advanced Robotics, Vol. 20, No. 6, pp. 707-736.

Lee, P. M., Hong, S.W., Lim, Y.K., Lee, C.M., Jeon, B.H., Park, J.W, 1999. Discrete-time quasi-sliding mode control of an autonomous underwater vehicle. IEEE Journal of Oceanic Engineering, 24 (3), 388-395.

Lee, P. M., Jeon, B. H., Lee, C. M., 2002. A docking and control system for an autonomous underwater vehicle, Proceedings of OCEANS 2002. MTS/IEEE, pp. 1609-1614.

Lee, P. M., Jeon, B. H., Kim, S. M., 2003. Visual servoing for underwater docking of an autonomous underwater vehicle with one camera, Proceedings of OCEANS 2003, Vol. 2, pp. 677-682.

Lee, P. M., Jun, B. H., Kim, K. H., Lee, J. H., Aoki, T., Hyakudome, T., 2007. Simulation of an inertial acoustic navigation system with range aiding for an autonomous underwater vehicle. IEEE Journal of Oceanic Engineering, 32 (2), 329-345.

Lee, P. M., Jun, B. H., 2007. Pseudo long base line navigation algorithm for underwater vehicles with inertial sensors and two acoustic range measurements, Ocean Engineering, Vol. 34, Issues 3-4, pp. 416-425.

Myring, D. F., 1976. A theoretical study of body drag in subcritical axisymmetric flow, Aeronautical Quarterly, vol. 27, pp 186-194.

Nerstein W., Smith, K.C., 1968. Hydrodynamic coefficient equations and computer programs. The bendix Corporation report.

Park, J. Y., Jun, B. H., Lee, P. M., Lee, F. Y., Oh, J. H., 2007. Experiment on underwater docking of an autonomous underwater vehicle 'ISiMI' using optical terminal guidance, Proceedings of OCEANS 2007 IEEE Aberdeen.

Prestero, T., 2001. Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle. M.S. Dissertation, MIT and WHOI.

Shapiro, L. G., Stockman, G. C, 2001. Computer Vision, Prentice Hall, NJ.

Singh, H., Bellingham, J.G., Hover, F., Lerner, S., Moran, B.A., Heydt, K., Yoerger, D., 2001. Docking for an autonomous ocean sampling network. IEEE Journal of Oceanic Engineering, 26 (4), 498-514.

Stokey, R., Allen, B., Austin, T., Goldsborough, R., Forrester, N., Purcell, M., Alt, C.V., 2001. Enabling technologies for REMUS docking : an integral component of an autonomous ocean-sampling network. IEEE Journal of Oceanic Engineering, 26 (4), 487-497.

Utkin, V.I., 1977. Variable structure system with sliding modes. IEEE Transactions on Automatic Control, 22 (2), 212-222.
http://auvlab.mit.edu/vehicles/vehiclespecEARLY.html#OD1
http://www.gavia.is/downloads/brochures/GaviaBrochure0402.pdf

21

Trajectory Planning for Autonomous Underwater Vehicles

Clement Petres[1], Yan Pailhas[2], Pedro Patron[2], Jonathan Evans[2], Yvan Petillot[2] and Dave Lane[2]
[1]*CEA-LIST,*
[2]*Heriot-Watt University, Ocean Systems Laboratory,*
[1]*France*
[2]*Scotland*

1. Introduction

1.1 Trajectory planning

This chapter is a contribution to the field of Artificial Intelligence. Artificial Intelligence can be defined as the study of methods by which a computer can simulate aspects of human intelligence (Moravec, 2003). Among many mental capabilities, a human being is able to find his own path in a given environment and to optimize it according to the situation requirements. For an autonomous mobile robot, the computation of a safe trajectory is crucial for the success of a mission. Here is the ultimate goal of the trajectory planning issue for autonomous robots:

given a set of internal and external constraints from the robot capabilities and from the environment
what is the best trajectory solution to reach a given target?

This is the problem we want to solve in this chapter. For this purpose, a novel approach is developed which is inspired from a level set method that originally emerged within the image processing community. This method, called Fast Marching (FM) algorithm, is analyzed and extended to improve the trajectory planning process for mobile robots. Theory and algorithms hold for any kind of autonomous mobile robot. Nonetheless, since this research work has been supported by the Oceans Systems Laboratory, the trajectory planning methods are applied to the underwater environment. Simulations and results are given assuming the use of an autonomous underwater vehicle (AUV).

1.2 Underwater environment and autonomous underwater vehicles

In mobile robotics, trajectory planning research has focussed on wheeled robots moving on surfaces equipped with high rate communication modules. The underwater environment is much more demanding: it is difficult to communicate because of low bandwidth channels undersea; it is prone to currents; and the three dimensional workspace may be worldwide. Moreover, torpedo-like vehicles are strongly nonholonomic.
The current state of technology allows many laboratories such as the Oceans Systems Laboratory to move forward in the development of AUVs. The need for a reliable cognition process for finding a feasible trajectory derived from underwater imagery is important.

1.3 Contributions

The main contribution of the authors is to present a Fast Marching based method as an advanced tool for underwater trajectory planning (Petres et al., 2007). With a similar complexity to classical graph-search techniques in Artificial Intelligence, the Fast Marching algorithm converges to a smooth solution in the continuous domain even when it is implemented on a sampled environment. This specificity is crucial to the understanding of the other contributions of our method:

- FM* algorithm: we develop a new algorithm called FM* that is a heuristically guided version of the Fast Marching algorithm. The FM* algorithm combines the efficiency of the A* algorithm (Hart, 1968) with the accuracy of the Fast Marching algorithm (Sethian, 1999).
- Curvature constrained trajectory planning: the FM* algorithm allows the curvature of the trajectory solution to be constrained, which enables us to take the turning radius of any mobile robot into account.
- Dynamic and partially-known domains: a dynamic version of the Fast Marching algorithm, called DFM, is proposed to deal with dynamic environments. DFM algorithm is then proved to be very efficient to recompute trajectories after minor changes in the robot perception of the world.
- Simulations and open water trials: a complete architecture has been designed, developed and tested for simulated and real AUV missions. In-water experiments are compared to simulation results to demonstrate the performance and usefulness of the DFM-based trajectory planning approach in the real world.

2. Trajectory planning framework

2.1 Environment representation

The usual framework to study the trajectory planning problem among static or dynamic obstacles is the *configuration space* (*C-space*). The main idea of the C-space is to represent the robot as a point, called a *configuration*.

A robot configuration is a vector of parameters specifying position, orientation and all the characteristics of the robot in the environment. The C-space is the set of all possible configurations. Its dimension is the number of parameters that defines a configuration. *C-free* is the set of configurations that are free of obstacles. Obstacles in the workspace become *C-obstacles* in the C-space.

Usually a simple rigid body transformation (Latombe, 1991) is used to map the real environment into the C-space. We focus on 2D and 3D C-spaces in this chapter, nonetheless this framework holds for C-spaces of any dimensions.

2.2 Problem statement

Given a C-space Ω, planning a trajectory is finding a curve

$$\begin{array}{rcl} C : [0,1] & \rightarrow & C - free \\ s & \mapsto & C(s) \end{array} \quad (1)$$

where [0,1] is the parameterization interval and s is the arc-length parameter of C. If x_{start} and x_{goal} are the start and the goal configurations respectively, then $C(0) = x_{start}$ and $C(1) = x_{goal}$.

An optimal trajectory is a curve C that minimizes a set of internal and external constraints (time, fuel consumption or danger for instance). It is assumed in this chapter that the complete set of constraints is described in a cost function τ:

$$\tau : \begin{matrix} \Omega \rightarrow \Re \\ x \mapsto \tau(x) \end{matrix} \tag{2}$$

2.3 Metric space

In this chapter the metric space Ω we refer to is the usual C-space equipped with the metric ρ defined as:

$$\rho(x_1, x_2) = \int_{[0,1]} \tau\left(C_{x_1, x_2}(s) \right) ds \tag{3}$$

where C_{x_1, x_2} is a trajectory between two configurations x_1 and x_2, and τ is the cost function. This metric can be seen as the "cost-to-go" for a specific robot to reach x_2 from x_1. At a configuration x, $\tau(x)$ can be interpreted as the cost of one step from x to its neighbours. If a C-obstacle in some region S is impenetrable, then $\tau(S)$ will be infinite. The function τ is supposed to be strictly positive for an obvious physical reason: $\tau(x) = 0$ would mean that free transportation from some configuration x is possible.

2.4 Distance function concept

A grid-search algorithm aims at building a distance function $u : \Omega^2 \rightarrow \Re$, which is solution of the functional minimization problem defined as follows:

$$u(x_{start}, x) = \inf_{\{C_{x_{start}, x}\}} \rho(x_{start}, x) \tag{4}$$

where $\{C_{x_{start}, x}\}$ is the set of all the possible curves between the source x_{start} and the current configuration x within Ω. For the sake of notational simplicity, and assuming that the source of exploration x_{start} is fixed, we note $u(x_{start}, x) = u(x)$.

The distance function u may be related to the *value function* concept in reinforcement learning. The difference lies only in the fact that value functions are refined in an iterative process (called learning), whereas the distance function is built from scratch. In the path planning literature one can find other names for the distance function, such as navigation function (LaValle, 2006), convex-map (Melchior et al., 2003) or multi-valued distance map (Kimmel et al., 1998).

Once the distance function has been found through the goal configuration, the optimal path is the one which follows the gradient descent over the distance function from the goal to the start configuration. This backtracking technique is reliable as no local minima have been exhibited during the exploration process.

3. Fast marching based trajectory planning

3.1 Related previous work

A method for computing consistent distance functions in the continuous domain was first proposed in (Tsitsiklis, 1995) but the method of the author is less efficient than the Fast

Marching method (Sethian, 1999). A FM based trajectory planning method among moving obstacles has been proposed in (Kimmel et al., 1998). The Fast Marching algorithm has also been applied in trajectory planning in (Melchior et al., 2003), where the authors compare A* and FM efficiencies among static obstacles. In (Philippsen & Siegwart, 2005), the authors develop a FM based trajectory planning method that allows dynamic replanning and improves Fast Marching efficiency in the case of a-priori unknown or dynamic domains. All these works are close in spirit to what we describe in this chapter except for the fact that we introduce a heuristic in a novel FM* algorithm to speed up the exploration process.

3.2 Eikonal equation

Before introducing the Fast Marching algorithm itself, we start from the observation that the functional minimization problem (4) is equivalent to solving the Eikonal equation:

$$\|\nabla u\| = \tau \tag{5}$$

We give here a geometrical intuition in two dimensions of how to convert equation (4) into equation (5). It is inspired by a level set formulation of the Eikonal equation in (Cohen & Kimmel, 1997) and a formal proof can be found in (Bruckstein, 1988).

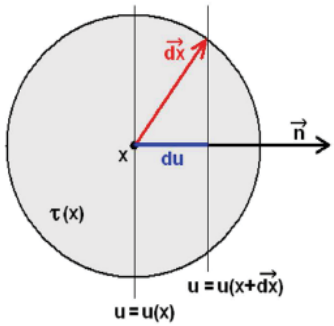


Fig. 1. On a small surface dΩ around a configuration x with a radius dx, one can approximate the distance function u as a plane wave, for which the level sets are parallel between them and perpendicular to the gradient ∇u of u.

We start from the fact that the gradient ∇u of u is normal to its level sets. Let $\vec{n} = \nabla u / \|\nabla u\|$, where $\|.\|$ is the Euclidean norm, be the outwards unit normal vector to level sets of u located in x (see figure 1). Express a variation du of u according to a variation $\overrightarrow{dx}$ of the position x:

$$u\left(x + \overrightarrow{dx}\right) = u(x) + \left\langle \nabla u, \overrightarrow{dx} \right\rangle$$

$$du(x) = \left\langle \nabla u, \overrightarrow{dx} \right\rangle \tag{6}$$

where $\langle .,. \rangle$ is the standard dot product in $\Re^2$.

Within the small region $d\Omega$ of Ω centered on x with a radius dx, we can assimilate τ as a constant: $\forall p \in d\Omega \; \tau(p) = \tau(x) = \tau$.

Within $d\Omega$ level sets of u are seen as straight lines:

$$du(x) = \tau\left\langle \vec{n}, \overrightarrow{dx} \right\rangle \tag{7}$$

From equations (6) and (7) we get $\left\langle \nabla u, \overrightarrow{dx} \right\rangle = \tau\left\langle \nabla u, \overrightarrow{dx} \right\rangle / \|\nabla u\|$, which leads to the Eikonal equation (5).

3.3 Upwind schemes and numerical approximations

The Fast Marching algorithm uses a first order numerical approximation of the Eikonal equation (5) based on the following operators. Suppose a function u is given with values $u_{i,j,k} = u(x_{i,j,k})$ on a 3D Cartesian grid with grid spacing h.

- Forward operator (direction i): $D^{+i}_{i,j,k}(u) = \left(u_{i+1,j,k} - u_{i,j,k}\right)/h$
- Backward operator (direction i): $D^{-i}_{i,j,k}(u) = \left(u_{i,j,k} - u_{i-1,j,k}\right)/h$

Forward and backward operators in directions j and k are similar.

The following upwind scheme, originally due to Godunov (Godunov, 1969) and well explained in (Rouy & Tourin, 1992) and in (Sethian, 1999), is used to estimate the gradient ∇u in three dimensions:

$$\begin{aligned} &\max\left(D^{-i}_{i,j,k}(u), D^{+i}_{i,j,k}(u), 0\right)^2 \\ &+ \max\left(D^{-j}_{i,j,k}(u), D^{+j}_{i,j,k}(u), 0\right)^2 = \tau^2_{i,j,k} \\ &+ \max\left(D^{-k}_{i,j,k}(u), D^{+k}_{i,j,k}(u), 0\right)^2 \end{aligned} \tag{8}$$

where $\tau_{i,j,k} = \tau\left(x_{i,j,k}\right)$.

3.4 Fast Marching algorithm

3.4.1 Pseudo-code

The pseudo code of the Fast Marching algorithm is given in table 1. The FM algorithm relies on a partitioning of the C-space in three sets: *Accepted* configurations for which the distance function u has been computed and frozen, *Current* configurations for which an estimate v of u has been estimated (and not frozen), and the remaining *Unvisited* configurations for which u is unknown.

Definitions

Start is the set of start configurations;
Goal is the set of goal configurations;
Neigh(S) is the set of neighbours of a set of configurations S;
x_{top} is the configuration in priority queue Current with the highest priority.

Procedure **Initialization()**

{01} Accepted = Start, u(Accepted) = 0;
{02} Unvisited = $\Omega \setminus$ Accepted, u(Unvisited) = v(Unvisited) = ∞;
{03} Current = Neigh(Start), v(Current) = τ(Current);

Procedure **Main()**

{04} Loop : while Goal $\not\subset$ Accepted
{05} Remove x_{top} from Current and insert it in Accepted with $u(x_{top}) = v(x_{top})$;
{06} FMComputeV(Neigh(x_{top}));

Table 1. Pseudo code of the Fast Marching algorithm

The set of Current configurations is stored in a priority queue. On top of this queue the configuration with the highest priority is called x_{top}. At each iteration of the exploration process, x_{top} is moved from Current to Accepted and its Unvisited neighbours are updated and moved from Unvisited to Current. The exploration process expands from the start configuration and ends when the goal configuration is eventually set to Accepted.

3.4.2 Computation procedure

The computation procedure for the 3D Fast Marching algorithm described in table 2 can be found in (Deschamps & Cohen, 2001). We give here additional calculation details to update the distance function estimate v_k of an x_{top}'s neighbour x_k with a cost τ_k.

Procedure **FMComputeV(Neigh(x_{top}))**
01} Loop : for all configurations x_k in Neigh(x_{top}) {02} If x_k is Unvisited, then remove it from Unvisited and insert it in Current with $v_k = \infty$ {03} If x_k is Current then apply case 1 or case 2 for the computation of v_k. {04} Sort Current list according to the priority assignment.

Table 2. Pseudo code of the FM procedure for updating Neigh(x_{top})

One, two or three Accepted configurations are used to solve equation (8). We note $\{A_1, A_2\}$, $\{B_1, B_2\}$ and $\{C_1, C_2\}$ the three couples of opposite neighbours of x_k (in 6-connexity) with the ordering $u(A_1) \le u(A_2)$, $u(B_1) \le u(B_2)$, $u(C_1) \le u(C_2)$ and $u(A_1) \le u(B_1) \le u(C_1)$. Two different cases are to be examined sequentially:

Case 1: considering that $v_k \ge u(C_1) \ge u(B_1) \ge u(A_1)$, the upwind scheme (8) is equivalent to:

$$(v_k - u(A_1))^2 + (v_k - u(B_1))^2 + (v_k - u(C_1))^2 = \tau_k^2 \tag{9}$$

Computing the discriminant of equation (9) there are two possibilities:

- if $\tau_k^2 > u(A_1)^2 + u(B_1)^2 + 2u(C_1)(u(C_1) - u(B_1) - u(A_1))$ and $u(B_1) < \infty$

$$v_k = \frac{1}{3}(u(A_1) + u(B_1) + u(C_1)) + \frac{1}{3}\sqrt{(3\tau_k^2 - 2(u(A_1)^2 + u(B_1)^2 + u(C_1)^2) - u(A_1)u(B_1) - u(A_1)u(C_1) - u(B_1)u(C_1))} \tag{10}$$

- else
  Go to case 2

Case 2: considering that $v_k \ge u(B_1) \ge u(A_1)$ and $v_k < u(C_1)$, the upwind scheme (8) is equivalent to:

$$(v_k - u(A_1))^2 + (v_k - u(B_1))^2 = \tau_k^2 \tag{11}$$

Computing the discriminant of equation (11) there are two possibilities:

- if $\tau_k > u(B_1) - u(A_1)$

$$v_k = \frac{1}{2}(u(A_1) + u(B_1)) + \frac{1}{2}\sqrt{2\tau^2 - (u(B_1) - u(A_1))^2} \tag{12}$$

- else

$$v_k = u(A_1) + \tau_k \tag{13}$$

Note that case 2 is similar to the update procedure of the 2D Fast Marching (Sethian, 1999).

3.5 FM* algorithm

In the Fast Marching algorithm the highest priority is assigned to the Current configuration x_{top} with the lowest estimate $e_{top} = v(x_{top})$, see table 3.

x_{top} (e_{top})	x_1 (e_1)	x_2 (e_2)	…	x_N (e_N)

Table 3. List of Current configurations stored in a priority queue. The highest priority is given the to lowest estimate e: $e_{top} < e_1 < e_2 < \ldots < e_N$.

Since u(x) does not depend on the goal configuration, the distance function u is built symmetrically around the start configuration, see figure 2.a. In this figure, distance maps and trajectories have been computed over a constant cost map. We use cool colours for small distances and hot colours for high distances (in arbitrary units).

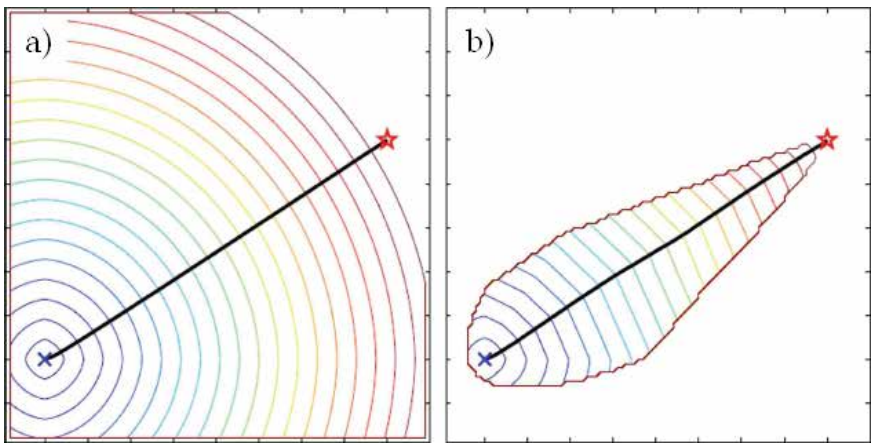


Fig. 2. Examples of distance maps and trajectories computed over a constant 100x100 cost map ($\tau = 1$) using a 4-connexity: a) FM algorithm and b) FM* algorithm (using the Euclidean distance H_e as a heuristic).

In the FM* algorithm the highest priority is assigned to the Current configuration x_{top} with the lowest estimate $e_{top} = \frac{1}{2} v(x_{top}) + \frac{1}{2} H_e(x_{top}, x_{goal})$. Here $H_e(x_{top}, x_{goal})$ is the *heuristic* that estimates the residual distance between the Current configuration x_{top} and the goal configuration x_{goal}. Similarly to the A* algorithm, instead of exploring around the start configuration, the FM* algorithm focuses the search towards the goal configuration, see figure 2.b.

Bi-directional versions of these grid-search algorithms can also be implemented. We just have to launch the grid-search algorithm simultaneously from the start and the goal configurations. We stop it when the two sets of Accepted configurations are merging.

4. Curvature constrained trajectory planning

In this section, differential constraints are reduced to curvature constraints. A Fast Marching based fully coupled approach (Petres et al., 2007) is proposed that ensures the trajectory solution to be smooth enough for an AUV with a given turning radius.

4.1 Problem statement

In this section the influence of the cost function τ on the smoothness of a trajectory C is analyzed. Here C is the solution of the functional minimization problem:

$$\begin{array}{c} \Omega^2 \to \\ (x_1, x_2) \mapsto \end{array} C = \underset{\{C_{x_1,x_2}\}}{\operatorname{argmin}}^{\tilde{\Omega}} \rho(x_1, x_2) \tag{14}$$

where $\tilde{\Omega}$ is the set of all the possible curves in Ω, $\{C_{x_1,x_2}\}$ is the set of all the possible curves in Ω between x_1 and x_2 and ρ is the continuous metric:

$$\rho(x_1, x_2) = \int_{[0,1]} \tau\left(C_{x_1,x_2}(s)\right) ds \tag{15}$$

The Fast Marching method computes a derivable solution C associated with the continuous metric ρ. Therefore, tools from differential geometry can be used to examine the curvature properties of C.

Let us define the curvature parameters considered here.

- Curvature magnitude of a curve C: $k(C) = \dfrac{\partial^2 C}{\partial^2 s}$
- Curvature radius of a curve C: $R(C) = \dfrac{1}{|k(C)|}$
- Lower bound on the curvature radius along a curve C: $R_{min}(C) = \inf_{s\in[0,1]} R(C(s))$
- Turning radius of a vehicle v: r(v)

4.2 Lower bound on the curvature radius

Given a cost function τ, our goal is to insure the feasibility of any trajectory C for an AUV v before computing the distance function u. Mathematically speaking, we want $\forall (x_1, x_2) \in \Omega^2, R_{min}(C) > r(v)$ knowing that $C = \underset{\{C_{x_1,x_2}\}}{\operatorname{argmin}} \rho(x_1, x_2)$. For this purpose we will express a formal link between the cost function τ and the lower bound $R_{min}(C)$ for any curve C minimizing the metric ρ between two configurations.

Using the differential geometry framework, it is shown in (Caselles et al., 1997) that the Euler-Lagrange equation associated with the functional minimization (14) is:

$$\tau k \vec{N} - \left\langle \nabla \tau, \vec{N} \right\rangle \vec{N} = 0 \tag{16}$$

where $\vec{N}$ is the normal unit vector to a curve C.

From equation (16), it is deduced in (Cohen & Kimmel, 1997) that the curvature magnitude k is bounded along any curve C minimizing ρ. The lower bound R_{min} is then:

$$R_{min} \geq \frac{\inf_{\Omega}\tau}{\sup_{\Omega}\{\|\nabla\tau\|\}} \tag{17}$$

The conclusion is that to increase the lower bound on the curvature radius $R_{min}(C)$ of an optimal trajectory C, two choices are possible:

- smoothing the cost function τ to decrease $\sup_{\Omega}\{\|\nabla\tau\|\}$
- adding an offset to the cost function to increase the numerator $\inf_{\Omega}\tau$ without affecting the denominator.

The following illustrations depict some trajectories computed using the FM* algorithm after smoothing the cost map (figure 3) and after smoothing the cost map and adding an offset (figure 4).

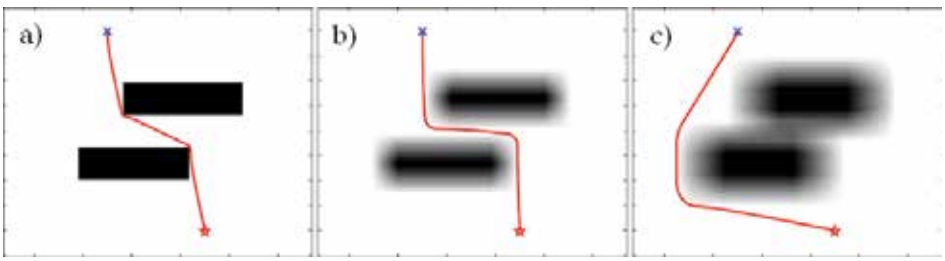


Fig. 3. Influence of smoothing the cost function. a) A binary 100x100 cost function τ, τ(C-free) = 1 and τ(C-obstacles) = 11 and the related optimal trajectory C_a, $R_{min}(C_a)$ = 332 (in arbitrary units). b) τ after smoothing using a 11x11 average filter, $R_{min}(C_b)$ = 1216. c) τ after smoothing using a 21x21 average filter, $R_{min}(C_c)$ = 1377.

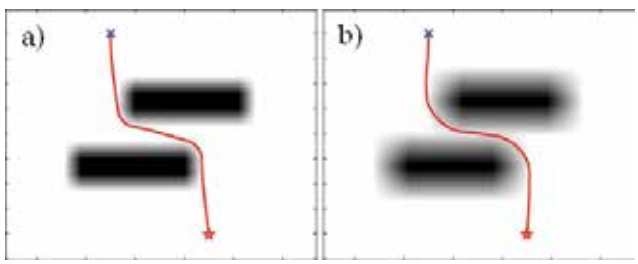


Fig. 4. Influence of both smoothing and adding an offset. The original cost function τ is similar to the one in figure 3.a. a) Offset = 5, average filter 7x7, $R_{min}(C_a)$ = 1977 (in arbitrary units). b) Offset = 5, average filter 15x15, $R_{min}(C_b)$ = 2787.

5. Trajectory planning in dynamic and partially-known domains

The two problems of planning trajectories in unpredictable dynamic environment and in partially-known environments are equivalent. In both cases the robot has to adapt its plans continuously to changes in (its knowledge of) the world. In this section we present a dynamic version of the Fast Marching algorithm called DFM and we compare it to A*, FM, FM* and D* Lite algorithms in simulated 2D environments. The DFM based trajectory planning method is eventually tested in a real open water environment using the AUV prototype of the Ocean Systems Laboratory.

5.1 DFM algorithm

The DFM algorithm is inspired from the LPA* and D* Lite algorithms described in (Koenig et al., 2004). It is similar to the E* algorithm developed by Philippsen in (Philippsen & Siegwart, 2005) but we prefer to name this algorithm DFM instead of E* because the asterisk usually refers to heuristically guided search algorithms (such as A* and D* algorithms). Since no heuristic has been integrated yet in any dynamic version of the Fast Marching algorithm, we propose to use the abbreviation DFM for Dynamic Fast Marching.
According to the principle of optimality it is not necessary to recompute an entire trajectory from A to B when a change appears in C somewhere between A and B. An efficient algorithm may only update the trajectory from C to B and leave the sub-trajectory from A to C unchanged.

5.1.1 Local consistency concept

Since changes appear dynamically in the cost function, any configuration $x \in \Omega$ may be updated more than once. The computation process of the distance function needs to be dynamic and the previous division between Accepted, Current and Unvisited sets of configurations is not compliant any more with a refresh of an Accepted configuration. Recall that an Accepted configuration in the static FM algorithm is frozen. Several updates of the estimate v(x) of the distance function u for a configuration x in Current is possible but the exploration process can only proceed forward from Unvisited to Accepted such as a flame in a landscape. The "engine" of this mechanism is that, at each iteration of the FM algorithm, the configuration x_{top} is moved from Current to Accepted. Then, its neighbours are updated, and the process continues until the goal configuration (initially tagged as Unvisited) is set as Accepted. The "Unvisited-Current-Accepted" scheme is well designed for static problems since a configuration can only proceed one way:

$$\begin{array}{ccccc} \text{Unvisited} & \rightarrow & \text{Current} & \rightarrow & \text{Accepted} \\ \infty & \rightarrow & v & \rightarrow & u \end{array}$$

In the DFM algorithm, the "tripartite" structure "Unvisited-Current-Accepted" is removed and replaced by a more subtle mechanism between the estimate v and the distance function u. The latter structure is made dynamic by the fact that the relationship between u and v is bilateral. The estimate v, which is affected by changes in the cost function τ, is computed from u, but u itself is computed from v:

$$\tau \rightarrow v \Leftrightarrow u$$

This mechanism, described in detail in the pseudo-code of the next section, stops when v and u match. The "engine" that leads to the "bipartite" agreement between v and u is the processing of a priority queue Q that contains exactly the inconsistent configurations defined as follows (Koenig et al., 2004). A configuration x is called *locally consistent* if $v(x) = u(x)$ and is called *locally inconsistent* if $v(x) \neq u(x)$. In (Philippsen & Siegwart, 2005), the authors reproduce this inequality in their pseudo-code. However, since Fast Marching methods use real numbers for approximating the distance function, a tolerance ε (set empirically at 0.1 in our implementations) must be introduced in the DFM algorithm so that the previous inequality becomes:

$$\left|v(x) - u(x)\right| > \varepsilon \tag{18}$$

5.1.2 Pseudo-code of the 3D DFM algorithm

The pseudo code of the 3D DFM algorithm is given in table 4.

```
Procedure CalculateKey(x)
{01} return [0.5*min(v(x), u(x)) + 0.5*He(x, xgoal); min(v(x), u(x))];
Procedure Initialize()
{02} Q = Ø;
{03} for all x ∈ Ω, v(x) = u(x) = ∞;
{04}      v(xstart) = 0
{05}      Q.Insert(xstart, [0.5*He(xstart, xgoal); 0]);
Procedure FMComputeV(x)
{06} Select configurations A1, B1, C1 using the computation procedure of table 2;
{07} Apply case 1 or case 2 using the computation procedure 3.4.2.
Procedure Update(x)
{08} if x ≠ xstart then v(x) = FMComputeV(x);
{09} if x ∈ Q then Q.Remove(x);
{10} if |v(x) - u(x)| > ε then Q.Insert(x, CalculateKey(x));
Procedure RunDFM()
{11} while Q.TopKey() < CalculateKey(xgoal) OR |v(xgoal) - u(xgoal)| > ε
{12}      x = Q.Pop();
{13}      if v(x) < u(x)
{14}           u(x) = v(x);
{15}           for all y ∈ Neigh(x) Update(y);
{16}      else
{17}           u(x) = ∞;
{18}           for all y ∈ Neigh(x)\{x} Update(y);
Procedure Main()
{19} Initialize();
{20} forever
{21}      RunDFM();
{22}      Wait for changes in τ;
{23}      for all configurations {x} with changed cost
{24}      Update τ({x})
{25}      Update({x})
```

Table 4. Pseudo code of the 3D DFM algorithm.

Main functions are:

- Q.Insert(x, key(x)): insert configuration x in the priority queue Q with priority key(x) = CalculateKey(x);
- Q.Remove(x): remove configuration x from the priority queue Q;

- Q.TopKey() = CalculateKey(x_{top})
- Q.Pop(): remove x_{top} from the priority queue Q and return it;

Main procedures are:

- **Initialize()**, lines {02-05}. Estimate v and distance function u are initialized at ∞, except for the start configuration x_{start} for which $v(x_{start}) = 0$. Then, start configuration is inconsistent and it is inserted in the priority queue Q described farther.
- **FMComputeV(x)**, lines {06-07}. The estimate v(x) is computed using the procedure described in table 2 similarly to the static 3D Fast Marching algorithm.
- **Update(x)**, lines {08-10}. First, v(x) is computed using FMComputeV(x). Second, x is removed from Q and, if x is still inconsistent, then it is re-inserted in Q.
- **RunDFM()**, lines {11-18}. The inconsistent configurations in Q are processed until their priorities become higher than the priority of the goal configuration x_{goal} AND x_{goal} becomes consistent (line {11}).

5.2 Application to trajectory planning for AUV in simulated environment

The purpose of this section is to test the DFM algorithm in a realistic simulated environment. First, a dynamic testbed is built, in which obstacles are supposed to be sensed by a sonar device. Second, DFM performance is analyzed and compared to some other dynamic trajectory planning algorithms.

5.2.1 Simulated testbed

We propose to use a simulated 500x500 pixels 2D sonar image as a cost function for testing the DFM algorithm. We want the cost function to be binary. It implies that we need to build a sonar image in which obstacles are supposed to be classified.

Before building the sonar image (SI), a binary map of obstacles (MO) is randomly generated. Three parameters control:

- the number of obstacles to generate ($nb_{Obst} = 50$),
- the number of obstacles to modify ($nb_{ObstMod} = 15$),
- the range of the width w_{Obst} and the length l_{Obst} of obstacles ($10 < w_{Obst}, l_{Obst} < 100$ (in pixels)).

The number of obstacles to be modified refers to the number of obstacles that will be randomly added or deleted from one map to another between the first and the second computation of the DFM algorithm in the tests of the next section.

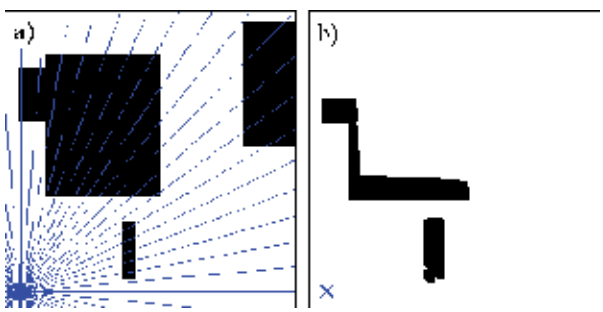


Fig. 5. Close-up on the computation of a simulated sonar image. a) A radial sweep is performed on a binary cost map using a virtual sonar beam (blue lines). b) The sonar echoes are interpreted to build the sonar image.

Once the map of obstacles has been generated, a ray tracing technique is used to build the sonar image. A radial sweep of 360 degrees is performed using a virtual sonar beam with a limited range (sonar-range = 150 pixels). It is assumed that obstacles have the properties of total reflection and homogeneous diffraction at the virtual frequency of the sonar, so that, when the beam meets an obstacle in MO, a spot is generated in SI (see figure 5). The size of the spot, which corresponds to the duration of the sonar pulse, is tuneable (size-spot = 10 pixels)

5.2.2 Dynamic trajectory planning optimization

In practice dynamic replanning algorithms are always launched from the goal to the robot configuration. These are the two reasons. First, it is obvious that dynamic trajectory planning algorithms are more efficient when changes appear close the goal location (Koenig et al., 2004). Since an AUV can only detect the changes that are close to its location (because of the reflection of the sonar beam by the obstacles and because of the limited range of the sonar), it is logic to consider the robot configuration as the goal configuration for the replanning algorithm.

Second, and more importantly, since the robot is continuously moving, its location is necessarily different between two consecutive replanning processes. If the robot configuration was considered as the start configuration of the replanning algorithm, the whole distance function should be permanently recomputed and the computational efficiency of the dynamic planning algorithm would be lost.

5.2.3 Comparative study

In this section a comparative study on a set of deterministic-sampling based dynamic trajectory planners is carried out to analyze the performance of the DFM algorithm. A*, FM, FM*, D* Lite and DFM algorithms are tested using the simulation testbed described in the previous section. The graph of figure 6 depicts the performance of the five trajectory planning algorithms over a range of replanning computations (runs).

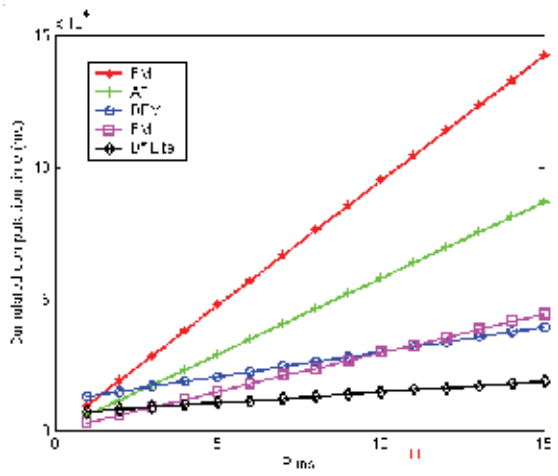


Fig. 6. Performance of A*, FM, FM*, D* Lite and DFM algorithms as a function of the number of runs. Each graph represents the evolution of the cumulated computation time of each algorithm over the runs.

One can see in figure 6 that FM* is the fastest static algorithm. However, from run 11, dynamic replanning algorithms (D* Lite and DFM) give better performance than static planning algorithms (A*, FM and FM*). This is explained by the greater efficiency of dynamic replanning algorithms when changes in the cost function happen close to the goal configuration.

5.2.4 Conclusion and future work

First, the FM* algorithm appears to be the best static trajectory planning algorithm (better than A*) both in terms of computation time and smoothness of the trajectory solutions. Second, dynamic trajectory planning algorithms are faster than static planners after a limited amount of time. Third, the DFM algorithm is slower than the D* Lite algorithm but it allows the curvature of the trajectories to be controllable.

Contrary to D* Lite, DFM algorithm does not include any heuristic to speed up the exploration process. One interesting direction for further research would be to develop what could be called the DFM* algorithm. A novel DFM* algorithm would combine the accuracy of the DFM algorithm with the exploration efficiency of the FM* algorithm.

5.3 Application to trajectory planning for AUV in real environment

In this section a complete AUV architecture designed to operate in unstructured environments is evaluated. Open water missions have been carried out to establish the performance of our FM based trajectory planning approach using the AUV prototype of the Ocean Systems Laboratory (Evans et al., 2008).

5.3.1 AUV architecture

It is important for an AUV to be able to follow complex scenarios and to rapidly respond to emergency situations. The following architecture has been designed to reach these objectives.

In this section both sensor and deliberative layers are described. Literature on actuators and control systems for AUVs may be found in (Hamilton et al., 2007; Fossen, 2002). The trajectory generation is provided by the DFM algorithm.

5.3.1.1 Sensor layer

The primary objective of the sensor layer is the generation of a local map. The output of this map provides an input for the deliberative layer that tries to match the arrangements of targets within the map against known scenarios. Our vehicle was equipped with inexpensive Tritech Sea King mechanically scanning forward looking sonar for obstacle detection. Navigation used an integrated GPS and Doppler Velocity Log solution mixing absolute and dead reckoning modes.

5.3.1.2 Deliberative layer

To provide deliberation in the generation of a safe behavior, a subsumption (Brooks, 1986) deliberative architecture has been chosen. It includes a reactive layer above a scenario layer as depicted in figure 7.

The reactive layer is empowered to take over from the scenario layer in the event of emergency, thus safeguarding the vehicle. It is deigned as a fuzzy system and it is triggered by range to nearest object.

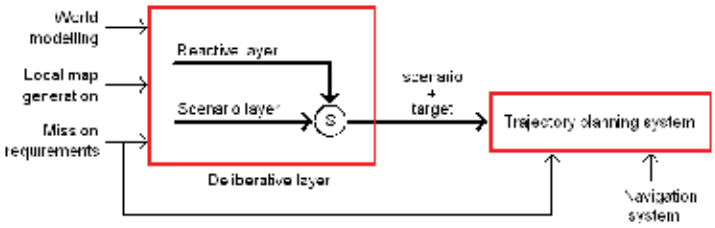


Fig. 7. Subsumption architecture implemented in the deliberative layer. A reactive layer is empowered to take over from the scenario layer in the event of emergency.

In the scenario layer, scripts called *scenarios* are employed. They are selected based on external and internal information along with mission requirements coming from the sensor layer. Ultimately, the deliberative layer sends the selected scenario and the selected target to the trajectory planning system, which generates the waypoints that are applied to the vehicle autopilot.

5.3.1.3 Trajectory generation

The trajectory planning method used in this module is based on the DFM algorithm. Since the local map around the vehicle is regularly updated, the DFM algorithm fulfils its real-time trajectory replanning mission.

5.3.2 Open water trials

A comprehensive set of open water trials have been carried out to validate some of the science reported above. Trials were carried out in Portmore Loch (Scotland) and Vobster Quarry Somerset (England) using HWU RAUVER hover capable AUV, see figure 8.a.

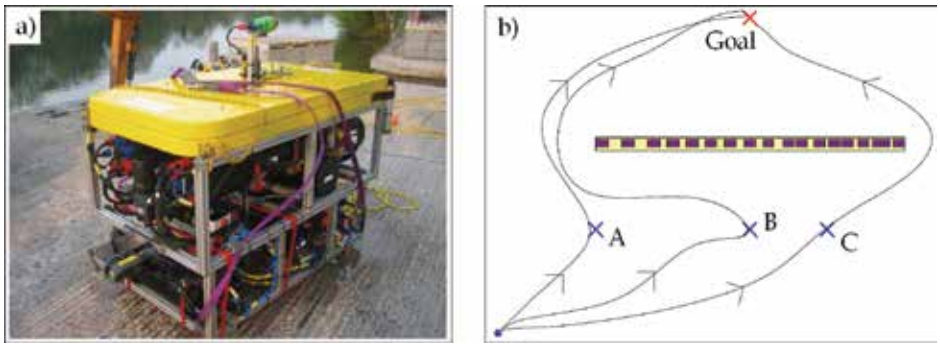


Fig. 8. The prototype of the Ocean Systems Laboratory, RAUVER, a hover capable autonomous underwater vehicle. b) In-water trials: net avoidance from different starting positions.

In this test, a net structure is set in the middle of the scene, see figure 8.b. The vehicle is sent to a waypoint located in the other side of the net. The starting point of the mission is situated in different places to observe the different behaviours. The expected behaviour is for the deliberative layer to plan a parallel course until the extent is detected, then a horizontal diversion. If this fails, the reactive layer should reverse to clear danger.
During the test starting from A the deliberative system found an alternative trajectory to reach the target avoiding the obstacle with a left horizontal diversion. In test B, the system,

that was keeping track of the next extension, found a trajectory on the left of the net and replanned the mission to go back and do the left horizontal diversion. Test C demonstrated that, by moving the starting point a little forward to the right of the net, extent is detected and the system is able to find a horizontal diversion on the right side of the net. In all cases, deliberative behaviours were always successful, without recourse to the reactive layer.

5.3.3 Conclusion

A complete architecture has been designed, developed and tested for real AUV missions. By inserting a scenario layer in the deliberative module, local maps generated in the sensor layer are used to choose and parameterize appropriate behaviours on the fly. A reactive layer has also been implemented and contributes by inhibiting goal points from the scenario layer in extremis. It has fuzzy behaviours to rapidly extricate the vehicle in case of emergency conditions. In practice, the scenario layer is rarely if ever inhibited by this reactive layer because the DFM based trajectory planning module produces safe trajectories.
Tests have been carried out on the real AUV prototype RAUVER of the Ocean Systems Laboratory in open water. Whilst the experiments reached the objectives, an unforeseen problem had to be solved: bottom reverberation. Bottom reverberation appeared as obstacle data and affected computed trajectories particularly at long ranges. In practice bottom reverberation has been detected and segmented into the local map, which successfully prevented these artefacts.

6. Conclusion

6.1. Recapitulative

The underwater world is a very demanding environment for trajectory planning algorithms. Great efforts are currently being made to develop autonomous systems as underwater technology becomes more mature. Several key issues for the three dimensional underwater trajectory planning problem have been addressed in this chapter. Reliability of trajectory planners has been improved by introducing the Fast Marching algorithm as a new basis for sampling based trajectory planning methods in the continuous domain.
First, we have introduced the trajectory planning framework and the basic concepts shared by all the deterministic sampling based planning algorithms. The Fast Marching method, as one of these trajectory planning technique is similar in spirit to classical grid-search algorithms such as the A* algorithm. This led us to develop a new algorithm, called FM*, that combines the exploration efficiency of the A* algorithm with the accuracy of the Fast Marching method. For these reasons, the FM* algorithm opens new possibilities for planning trajectories in wide and continuous underwater environments.
Second, even if they are implemented on a discretized perception of the world, Fast Marching based planning methods have the property to extract derivable trajectories. By applying mathematical tools from differential geometry, it has been proved that smoothing input data results in smoother trajectories. A technique has been proposed that insures the feasibility of a trajectory for a mobile robot with a given turning radius. This technique iteratively smoothes input data until a formal criterion is satisfied. The method is efficient because the Fast Marching algorithm is eventually launched only when input data are compliant with the curvature constraints of the vehicle.
Third, another approach has been developed to speed up the exploration process in the case of partially-known or dynamic environments. A dynamic version of the Fast Marching

algorithm, called DFM, has been presented that is able to reuse information of previous searches. Compared to A*, FM, FM* and D* Lite algorithms, the DFM algorithm is very efficient when changes happen randomly in the vehicle's perception of the world.
Eventually, a complete architecture has been designed, developed and tested for real AUV missions. Performance and usefulness of the DFM based trajectory planning approach in partially-known domains have been demonstrated using the experimental prototype of the Ocean Systems Laboratory.

6.2 Future work

6.2.1 High dimensional state spaces

Even if the O(Nlog N) complexity of the Fast Marching algorithm is similar to the complexity of classical discrete grid-search algorithms, FM based trajectory planners are suitable for C-spaces with only a few numbers of dimensions (at most three in practice). The DFM algorithm improves re-planning efficiency of trajectory planners in unpredictable or a priori unknown environments. Nonetheless, further research would benefit from the addition of a heuristic to the DFM algorithm in order to speed up its exploration capacities.

6.2.2 Planning with uncertainty

In this chapter we have not dealt with uncertainties on the AUV perception of the world. It has been assumed that the vehicle had either an a priori comprehensive knowledge of its environment (chapters 2, 3 and 4) or had a limited visibility (chapter 5). In both cases, precise location of C-obstacles was assumed. This hypothesis is not very realistic as underwater sensors have limited performance. In (Petres, 2007) a dilation of C-obstacles is proposed to improve the safety of the trajectories. This simple method is easy to implement practically but it does not explicitly include uncertainties about sensor specifications. Further work on FM based trajectory planning for AUV using an information space representation would be promising.
On the other hand, uncertainties about AUV configurations have not been considered in the presented trajectory planning methods. This is not very realistic as accurate underwater navigation is still a challenging issue. Nonetheless, concurrent mapping and localization (CML) techniques exist for AUV navigation (Tena Ruiz et al., 2004). This topic is out of the scope of this chapter but further development would benefit from a joint navigation approach for AUV that would couple CML and advanced trajectory planning techniques.

7. References

Brooks, R.A. (1986). A Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation*, Vol. 2, No. 1, pp. 14-23, ISSN: 0882-4967

Bruckstein, A.M. (1988). On shape from shading, *Computer Vision, Graphics, and Image Processing*, Vol. 44, No. 2, pp. 139-154, ISSN: 0734-189X

Caselles, V.; Kimmel, R. & Sapiro, G. (1997). Geodesic Active Contours, *International Journal of Computer Vision*, Vol. 22, No. 1, pp. 61-79, ISSN: 0920-5691 (Print), 1573-1405 (Online)

Cohen, L.D. & Kimmel, R. (1997). Global Minimum for Active Contour Models: A Minimal Path Approach, *International Journal of Computer Vision*, Vol. 24, No. 1, pp. 57-78, ISSN: 0920-5691 (Print), 1573-1405 (Online)

Deschamps, T. & Cohen, L.D. (2001). Fast Extraction of Minimal Paths in 3D Images and Applications to Virtual Endoscopy, *Medical Image Analysis*, Vol. 5, No. 4, pp. 281-299

Evans, J.; Patron, P.; Smith, B. & Lane, D.M. (2008). Design and Evaluation of a Reactive and Deliberative Collision Avoidance and Escape Architecture for Autonomous Robots, *Autonomous Robots*, Vol. 24, No. 3, pp. 247-266, ISSN: 0929-5593

Fossen, T.I. (2002). *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs, and Underwater Vehicles*, Marine Cybernetics, ISBN: 82-92356-00-2

Godunov, S.K. (1969). A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations, *Sbornik Mathematics*, Vol. 47, pp. 271-306

Hamilton, K.; Lane, D.M.; Brown, K.E.; Evans, J. & Taylor, N.K. (2007). An Integrated Diagnostic Architecture for Autonomous Underwater Vehicles: Research Articles, *Journal of Field Robotics*, Vol. 24, No. 6, pp. 497-526, ISSN: 1556-4959

Hart, P.E.; Nilsson, N.J.; Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Transactions on Systems Science and Cybernetics*, Vol. 4, No. 2, pp. 100-107, ISSN: 0536-1567

Kimmel, R.; Kiryati, N. & Bruckstein, A.M. (1998). Multi-Valued Distance Maps for Motion Planning on Surfaces with Moving Obstacles, *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 3, pp. 427-436, ISSN: 1042-296X

Koenig, S.; Likhachev, M.; Liu, Y. & Furcy, D. (2004). Incremental Heuristic Search in Artificial Intelligence, *Artificial Intelligence Magazine*, vol. 25, No. 2, pp. 99-112, ISSN: 0738-4602

Latombe, J.-C. (1991). *Robot Motion Planning*, Kluwer Academic Publisher, ISBN: 079239206X, Norwell, MA, USA

LaValle, S.M. (2006), *Planning Algorithms*, Cambridge University Press, ISBN-10: 0521862051, ISBN-13: 978-0521862059

Melchior, P.; Orsoni, B.; Lavialle, O.; Poty, A. & Oustaloup, A. (2003). Consideration of Obstacle Danger Level in Path Planning Using A* and Fast-Marching Optimization: Comparative Study, *Signal Processing*, Vol. 83, No. 11, ISSN: 0165-1684

Moravec, H. (2003). Robots, After All, *Communications of the ACM*, Vol. 46, No. 10, pp. 90-97, ISSN: 0001-0782

Petres, C.; Pailhas, Y.; Patron, P.; Petillot, Y.; Evans, J. & Lane, D.M. (2007). Path Planning for Autonomous Underwater Vehicles, *IEEE Transactions on Robotics*, Vol. 23, No. 2, pp. 331-341, ISSN: 1552-3098

Petres, C. (2007). *Trajectory Planning for Autonomous Underwater Vehicles*, Heriot-Watt University, Ph.D. Thesis

Philippsen, R. & Siegwart, R. (2005). An Interpolated Dynamic Navigation Function, *Proceedings of IEEE Conference on Robotics and Automation (ICRA 2005)*, pp. 3782-3789, ISBN: 0-7803-8914-X

Rouy, E. & Tourin, A. (1992). A Viscosity Solutions Approach to Shape-from-Shading, *SIAM Journal on Numerical Analyzis*, Vol. 29, No. 3, pp. 867-884, ISSN: 0036-1429

Sethian, J.A. (1999). *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, ISBN: 0521645573, 9780521645577, Cambridge, MA, USA

Tena Ruiz, I.; de Raucourt, S.; Petillot, Y. & Lane, D.M. (2004). Concurrent Mapping and Localization Using Sidescan Sonar, *IEEE Journal of Oceanic Engineering*, Vol. 29, No. 2, pp. 442-456, ISSN: 0364-9059

Tsitsiklis, J.N. (1995). Efficient Algorithms for Globally Optimal Trajectories, *IEEE Transactions on Automatic Control*, Vol. 40, No. 9, pp. 1528-1538, ISSN: 0018-9286

22

Environmental Impact Assessment and Management of Sewage Outfall Discharges using AUV'S

Patrícia Ramos[1,2] and Mário V. Neves[2]
[1]*Institute of Accountancy and Administration of Porto*
[2]*Faculty of Engineering of University of Porto*
Portugal

1. Introduction

Wastewaters are often discharged into coastal waters through outfall diffusers that efficiently dilute effluent and usually restrict any environmental impact to a small area. Predicting this impact is however difficult because of the complexity of the hydrodynamic processes that mix the wastewater and also because of the variability in oceanic conditions.
Despite great improvements in the understanding of these mixing processes over the years, since models are now available that can make reasonable predictions under steady-state conditions, there remain many aspects unknown and unpredictable. For this reason, much effort has been devoted recently to improving means to monitor and characterize sewage plumes under a variety of oceanographic conditions.
Traditional sampling techniques for underwater observation are generally expensive and do not offer a comprehensive coverage, specially as the requirements of oceanographic and environmental field studies, as those considered here, become more and more demanding. Clearly, *in situ* platforms, such as Autonomous Underwater Vehicles (AUVs), which are cost effective, mobile, and capable of capturing key phenomena adaptively, which is particularly interesting for oceanographic processes that vary widely and over several time and length scales, as is the case of sewage plumes, may improve environmental monitoring and predictive modeling for decision-making and management (Fletcher, 2001; Robinson et al., 1999; Yu et al., 1994).
In this chapter we describe how AUVs can be used with advantage for environmental impact assessment and management of sewage outfall discharges. The details and results of a monitoring campaign to a Portuguese outfall using an AUV are fully described.

1.1 Isurus AUV

Isurus is a REMUS class AUV, acquired to the Woods Hole Oceanographic Institution, MA, USA, in 1997 (Fig. 1). These vehicles are low cost, lightweight AUVs specially designed for coastal waters monitoring (Alt et al., 1994). The reduced weight and dimensions make them extremely easy to handle, requiring no special equipment for launching and recovery.
Isurus has a diameter of 20 cm and is about 1.5 m long, weighting about 35 kg in air. The maximum forward speed of the vehicle is 2 m/s, however the best energy efficiency is

achieved at about 1 m/s. At this velocity, the energy provided by a set of rechargeable lithium-ion batteries may last for over 20 hours.

Fig. 1. *Isurus* Autonomous Underwater Vehicle.

Although small in size, this vehicle can accommodate a wide range of oceanographic sensors, according to mission objectives. For the field experiment described here two specific sensors were integrated: a conductivity, temperature, depth (CTD), OS200 model from Ocean Sensors, Inc., USA, and an altimeter to measure the height from the sea bottom, from Imagenex, Inc., Canada.

Since this type of vehicle operates without external control to perform the pre-defined mission transects, they have to be able to locate themselves in the field. The navigation system of *Isurus* AUV estimates in real time its position based on data provided from several devices and systems. The vertical coordinate is obtained from a pressure cell installed on the vehicle. To estimate the horizontal position, the navigation system fuses together dead reckoning data with absolute positioning data (Matos et al., 1999).

Typically, a mission starts with the vehicle programming, by editing a mission file in the laptop computer, with the tasks that need to be successively accomplished and the exact field location of the navigation transponders. After the final diagnostic, where the status of the various subsystems is checked, the mission file is transferred to the vehicle and the cable is disconnected. The vehicle is then launched in the water, from the coast or from a small boat. At this stage, the navigation and control programs are running and dictate the vehicle manoeuvres throughout the mission. The mission ends with the recovery of the vehicle in the expected final position. The connection with the laptop is then re-established and all the stored information is downloaded.

There are two data sets collected during a typical mission. The first set is related to the vehicle internal data, which are only analysed in the laboratory. These allow to evaluate the performance of various subsystems (navigation, control, power consumption, etc) and provide useful information for potential improvements. The other data set comprises all the information from the oceanographic sensors.

Usually, before processing, it is necessary to locate the collected data in time and space. While time location is obtained directly from the vehicle real-time clock, the spatial location is based on the vehicle position estimate provided by the navigation system (horizontal coordinates) and the depth cell (vertical coordinate).

If the spatial location of oceanographic data gathered with an AUV is known in a relatively precise manner, the same cannot be said for towyo platforms. In this case, even if the research vessel location is perfectly known, the natural instability of the cable, very difficult to control in a precise way specially with increased depth, can lead to positioning errors of the towed sensor package which can be critical for small scale phenomena observation.

1.2 S. Jacinto outfall

A map of the study area is shown in Fig. 2. The S. Jacinto outfall is located off the Portuguese west coast near the *Aveiro* estuary. The total length of the outfall, including the diffuser, is 3378 m (the first 3135 m section having a diameter of 1600 mm and the last 243 m section having a diameter of 1200 mm). The diffuser consisting of 72 ports alternating on each side, nominally 0.175 m in diameter, is 332.5 m long. Currently, only the last 20 of the 72 ports are working in a length of 98.2 m. These are discharging upwards at an angle of 30º to the horizontal axis, being the port height about 1.3 m. The outfall has a true bearing direction of 290º and is discharging at a varying depth approximately between 14 and 17 m.

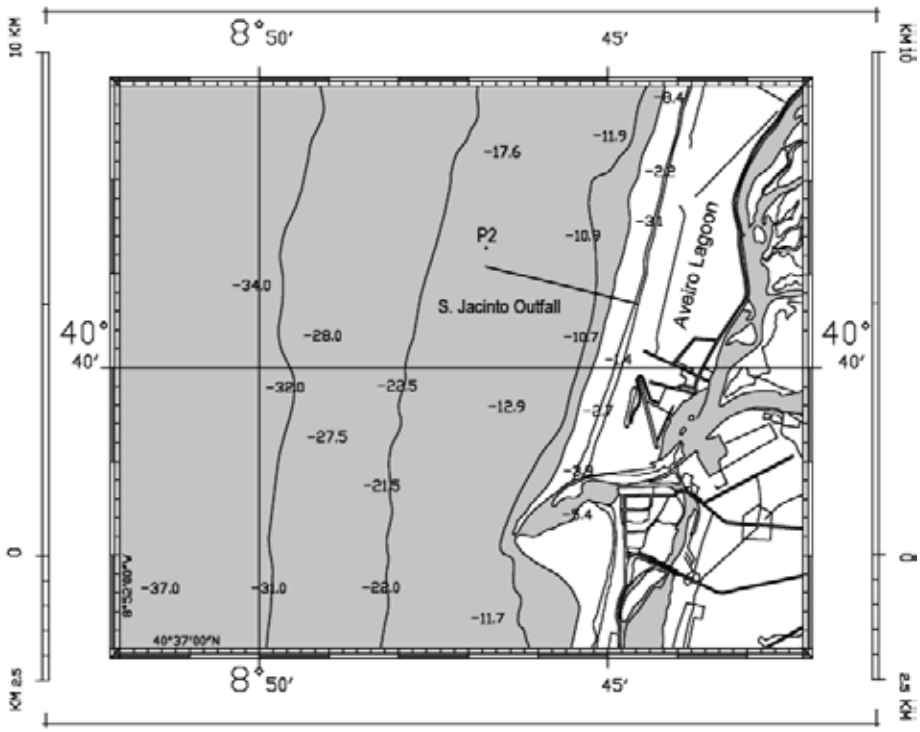


Fig. 2. Map of the study area.

The sea floor near the diffuser has a moderately sloping, sandy bottom with isobaths oriented parallel to the coastline. In that area the coastline itself runs at about a 200º angle with respect to true north. Flow variation through the outfall in question is not typical of WWTPs since the effluent is mainly of industrial origin. Effluent flowrate ranges most frequently between 0.6-0.8 m^3/s.

During the campaign the discharge remained fairly constant, with an average flowrate of approximately 0.61 m^3/s.

The remainder of the chapter is outlined as follows. Section 2 describes the AUV near field monitoring mission to *S. Jacinto* outfall, focusing on the near-field transects and the ambient conditions. Data processing and results in terms of plume dispersion and dilution estimation are presented in Section 3. Finally, a brief summary of the main conclusions is presented in Section 4.

2. AUV monitoring mission

One of the major problems related to the plume tracking studies is the sampling plan (Ramos, 2005). Usually, by observation of typical conditions in the vicinity of the outfall, namely density stratification and primary current directions that are most likely, investigators have some expectations of the effluent plume direction when it leaves the diffuser, its height of rise, thickness and even dilution.
However, oceanographic conditions seldom correspond exactly to those expected and values of density stratification, current speed and direction, etc will almost certainly be different in some degree from those predicted. When there is no flexibility in the plume tracking work plan, these changes in the expected oceanographic conditions can cause some problems: for example; (1) higher current speeds may result in an effluent plume that is more dispersed and that may extent beyond the survey area; (2) field transects not perpendicular to along-current direction may cause a distortion of plume characteristics such as spreading width, length of hydrodynamic mixing zone and minimum dilution, not valid to be compared with e.g. RSB model (Roberts et al.,1989) predictions which are to be observed in a vertical plane through the wastefield, perpendicular to current direction, at the end of the initial mixing region; (3) transects shape and resolution of the transects do not permit the end of the near field to be distinguished (Roberts et al., 2002); (4) the prescribed survey area may provide sampling out of the wastefield.
A race track design or any other type of complete sequential loops around the diffuser does not miss the plume, but, besides gathering useless data, is certainly not optimal for minimizing spatial and temporal aliasing effects.
Conducting previous exploratory transects to find gradients in the field that help to specify plume structure and behaviour may become a difficult task to natural tracer tracking.
Minimum dilutions reported in field studies may not always have been measured at the end of near field. In some cases, as admitted by the authors, measurements may have been obtained in a still rising plume, in which case dilution was incomplete. When the expected near field sampling actually took place beyond the hydrodynamic mixing zone, upper dilutions were certainly considered.

2.1 Mission transects

Within the near field, plumes are energetically turbulent with wide variations in instantaneous measurements (Roberts, 1996). Sampling within this area is therefore problematic as results may fluctuate widely and be difficult to interpret. One of the main goals of this work was to develop a reliable *in situ* method to monitor pollutants discharged in the environment with increased temporal and spatial resolution, making use of an AUV.
In order to reduce the uncertainty about plume location and concentrate the monitoring mission only in the hydrodynamic mixing zone, outputs of a near field prediction model, based on effective real time *in situ* measurements of current speed and direction and density stratification, were opportunistically used to specify in real time the mission transects (Ramos, 2005).
The near field model used was RSB (Roberts et al., 1989). The RSB model, based on the experimental work on multiport diffusers of Roberts et al. (1989) using dimensional analysis and length scale arguments, incorporates the most important hydrodynamic aspects of ocean outfalls. These include, the effects of arbitrary current speed and direction (including parallel currents), stratification, port spacing, source momentum flux, discharges from both

sides of the diffuser and the resulting merging of the plumes from both sides, re-entrainment and additional mixing in the spreading layer, direct plume impingement in parallel currents, and lateral gravitational spreading.

The model has been validated against data: (1) from a field tracer and laboratory experiments to the San Francisco outfall (Roberts & Wilson, 1990), (2) from field tests to Whites Point, Los Angeles outfall (Washburn et al., 1992) and Sand Island, Hawaii outfall (Petrenko et al., 1998), and recently (3) from field tracer experiments to Boston outfall (Roberts et al., 2002).

The RSB outputs, including length of hydrodynamic mixing zone, spreading width at the end of the near field, maximum rise height and thickness, in conjunction with current direction, were used to define the sampling transects.

To guarantee the plume observation at the end of near field, the longitudinal distance was actually considered somewhat greater than the initial mixing zone length. The maximum downstream normalized distance used by Roberts et al. (1989) in its laboratory experiments was, in fact, the survey length considered. A minimum safe distance to the outfall was also guaranteed to avoid jets turbulence affect the vehicle navigation. The minimum downstream normalized distance used by Roberts et al. (1989) in its laboratory experiments was used as reference. The wastefield width was increased in 20% to take into account experimental errors, as suggested by Roberts et al. (1989). A minimum distance of 4 m from the sea bed (due to safety navigation requirements), and a minimum distance of 2 m from the surface (due to wave motions interference with the vehicle navigation) was also considered. Eliminating unnecessary measurements, this adaptive sampling approach enables to increase the horizontal and vertical resolutions, that are specially critical to natural tracer tracking in environments with large gradients in background values of the natural tracer.

Yo-yo shaped transects were not particularly useful in this case. With a maximum dive angle of 10^{o}, the vehicle would perform minimum cycles of 159-193 m in yo-yos of 14-17 m amplitude, with a horizontal resolution between 79-96 m at the middle of the water column. If the wastefield width at the end of near field is about 1.5 times the diffuser length (Roberts et al. 1989), i.e. 147 m, the AUV would perform not more than 2 yo-yos in each transect, missing lot of data.

So, horizontal transects at different depths with a minimum vertical spacing of 2 m and a minimum spacing of 20 m between horizontal parallel transects, due to the natural navigation variability in depth of *Isurus* AUV, seemed to be a right choice for these specific study conditions.

Tracks perpendicular to current direction, instead of parallel, are preferred to minimize temporal aliasing between samples at the same cross section. Cross sections at the end of near field can be comparable to RSB model predictions. Tracks parallel to down current direction are preferred to observe how dilution varies with distance.

To be able to compare field observations with model predictions, density vertical profiles should be collected outside the dispersion area (unaffected by the plume salinity). Those profiles are necessary to run the models.

Two applications were developed to implement easily in the field the sampling strategy adopted. *Sea Outfall Monitoring Campaign GUI* is used for the automatic specification of the inspection area, and *Isurus Mission GUI* is used for the automatic definition of the AUV mission file (see their layouts during S. Jacinto outfall campaign in Fig. 3 and Fig. 4).

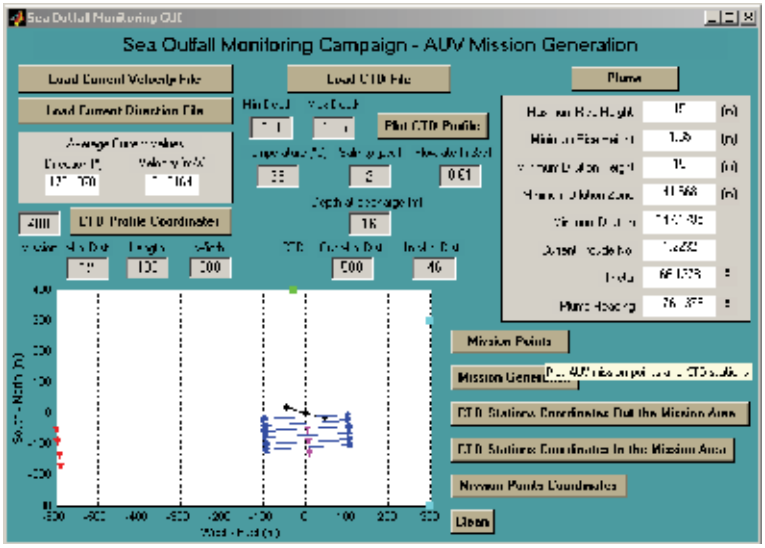


Fig. 3. *Sea Outfall Monitoring Campaign GUI* layout during for S. Jacinto outfall monitoring campaign.

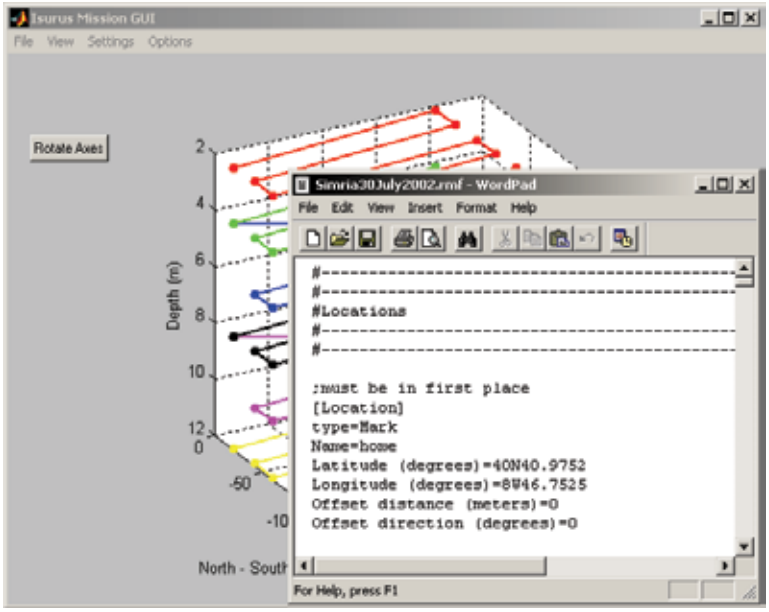


Fig. 4. *Isurus Mission GUI* layout during for S. Jacinto outfall monitoring campaign.

According to RSB model predictions, obtained in real time during S. Jacinto outfall monitoring campaign, the plume was spreading at the surface, detached from the bottom and forming a two-layer flow. At the end of the near field 42 m downstream from the diffuser, the predicted wastefield width was about 177 m.

The AUV monitoring mission took about 112 minutes, starting approximately at 14:00 GMT. A rectangular area of 200 m x 100 m starting 20 m downstream from the middle point of the diffuser was covered (see the vehicle position estimate in Fig. 5).

As predicted, the vehicle performed six horizontal trajectories at 2, 4, 6, 8, 10, and 12 m depth. In each horizontal trajectory, the vehicle described six parallel transects perpendicular to the water current direction, of 200 m length and spacings of 20 m (labelled as section 1-6 in Fig. 5).
In performing horizontal trajectories, vertical oscillations of the AUV were less than 0.5 m.
During the mission the vehicle transited at a fairly constant velocity of 1 m/s.
CTD data were recorded at a rate of 2.4 Hz, so horizontal resolution was about 0.4 m (horizontal resolution is defined here as the approximate distance between consecutive points that are sampled at the same depth, i.e., in the same layer). Vertical resolution varied along the mission due to the influence on the vehicle navigation of natural currents but was almost always between 1-2 m (vertical resolution is defined as the vertical distance between points in the water column that are sampled approximately at the same (X,Y) location but at consecutive depths, i.e., on successive layers).
Comparing to other field studies (Washburn et al., 1992; Wu et al., 1994; Petrenko et al., 1998) this resolution corresponds to a huge improvement in plume tracking surveys.

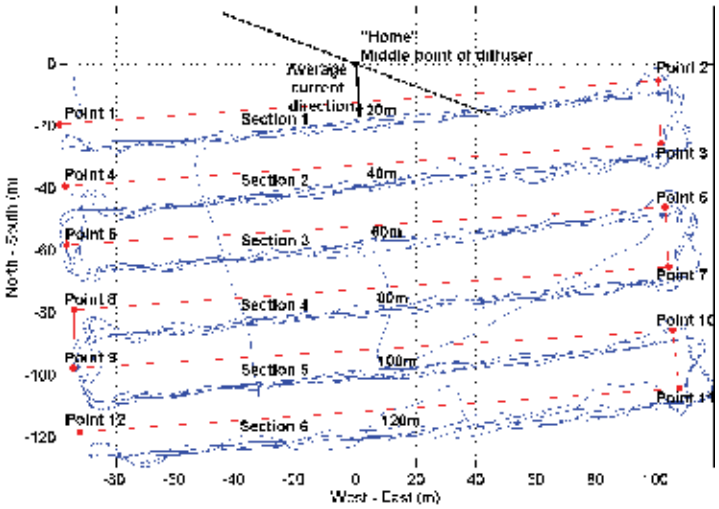


Fig. 5. Plan view of the AUV position estimate during the plume tracking survey downstream from the outfall diffuser.

Although the position estimate, provided by the navigation system, is accurate enough for the guidance and control of the AUV, it is not well suited for spatially locating the data collected. In fact, the discrete nature of the correction mechanism used in real time gives rise to discontinuities in the evolution of the estimate, which is a major drawback for spatially locating the collected data (Matos et al., 2003).
An algorithm that processes the data collected by the vehicle navigation system during the execution of a mission and produces an estimate of the trajectory described by the vehicle more suited for spatially locating the oceanographic data gathered was applied (see the details in Matos et al. (2003)). The final result is shown in Fig. 5.
Little deviation from the line transects specified (in red) is observed, with an exception to the trajectory at 4 m depth, in sections 3 and 6, where the navigation system could not correct the vehicle position estimate, because no valuable information was being received at that time from the transponders.

A larger distance of ~20 m from the middle point of the diffuser to the middle point of section 1, instead of the 12 m considered in the mission specification, was observed. This offset of about 8 m between the predicted and performed trajectories was not result of a poor performance of the navigation algorithm, but a direct consequence of the relative strong water current to the south during the mission. In fact, the heading reference generated by the vehicle guidance system does not integrate the off track error, giving rise to large off track errors when there are large water currents perpendicular to the desired trajectory (Matos et al. 2003).
During the vehicle deployment process, a drift from the "home" position, where it was supposed to be launched, occurred, so a correction to the expected trajectory from Point 1 to Point 2 was efficiently performed by the navigation system at the beginning of the mission in the trajectory at 2 m depth.

2.2 Field conditions summary

The data on the currents field were obtained from a downward-looking Acoustic Doppler Current Profiler (ADCP) (Workhorse Sentinel 600 kHz model, RD Instruments, Inc.) deployed through a small boat for a few minutes before the start of the AUV mission.
One vertical profile of horizontal current components was provided with measurements at 0.5 m vertical increments between 1.55 and 14.05 m depth.
The motion of the surface waters, down to a depth of about 8.55 m, was generally to the South; the deeper currents moved generally between SE-SSE, with average direction 176.14º over the whole water column. The predominant current direction was then to the south, almost perpendicular to the diffuser axis. Current speeds were generally between 0.1-0.25 m/s over the whole water column, with the average being about 0.132 m/s.
Information on density stratification at the same location was also obtained with an OCEANSEVEN 316 multi-sensor (manufactured by IDRONAUT-Italy) to run RSB model.
A similar profile was acquired by *Isurus* AUV close to Point 1 during its ascending trajectory from 12 m depth to the surface, at the end of the mission. Fig. 6 shows vertical temperature, salinity and density measurements averaged into 2-s bins.

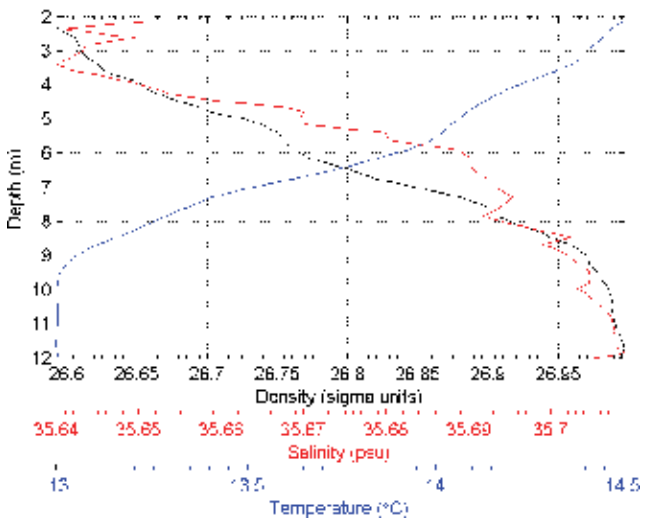


Fig. 6. Vertical STD profiles measured by *Isurus* near the diffuser.

The water column was weakly stratified due to both low temperature and salinity variations in the vicinity of the diffuser.
The temperature difference between the surface and the bottom was about 1.5ºC, with most of the difference occurring in a thermocline that extended from ~2 m to 10 m.
The salinity profile showed some variability extending over the whole water column, with the difference between the bottom and the surface being about 0.07 psu.
The total difference in density over the water column was about 0.48 σ-units, with the change with depth being approximately linear.

The buoyancy frequency from the discharge depth to the surface was then

$$N = \sqrt{-\frac{g}{\rho_a}\frac{d\rho}{dz}} = 0.02 s^{-1}, \tag{1}$$

where $g = 9.8$ is the acceleration due to gravity, $\rho_a = 1026.99$ is the seawater density at the port depth, and $d\rho/dz = 0.48/15$ is the vertical density gradient. The discharge depth during the AUV mission was about 15 m. The Froude number F, based upon the current speed and one of the most important parameters governing the dynamics of the diffuser flow in the near-field, was equal to (Roberts et al., 1989)

$$F = \frac{u^3}{b} = 1.18, \tag{2}$$

where $u = 0.132$ is the current speed, and b is the buoyancy flux per unit length of the diffuser equal to $g(\Delta\rho/\rho_a)Q/L$, where $\Delta\rho = 31.47$ is the density difference between the seawater and wastewater at the port level, $Q = 0.61$ the volume flow rate of wastewater, and $L = 98.2$ the diffuser length.

3. Results and discussion

3.1 Plume spatial characterization

3.1.1 Data processing

In order to map effectively the dispersion of the effluent using the AUV data, three main steps were followed.
In the first step, after a global analysis of the collected data (where, for example, errors due to sensor malfunctions were detected), an estimate of the appropriate trajectory described by the vehicle for the spatial location of the collected oceanographic data was produced.
The uncertainty of the horizontal position estimate was less than 5 m. The uncertainty of the vertical position estimate was even less, due to the high precision complementary measurements of the eco-sounder. Then CTD and navigation data were merged onto a common time base using linear interpolation. Positioning data were rotated about -3.86º so that North-South/East-West lines were aligned with $x - y$ axis.

In the second step, using the polynomials developed by Millero et al. (1980), *in situ* conductivity, temperature and pressure were used to compute salinity. Then, density was estimated by using this computed salinity and the measured temperature and pressure. Finally, the last step was to plot the desired variables onto $x - y$, $x - z$, and $y - z$ $ grids using the Least Squares Collocation Method (LSCM) technique (see details in Zhang et al.

(2001) and Ramos (2005)). The LSCM has been used for numerous applications, namely the numerical solution of differential equations such as the Navier-Stokes equations and hyperbolic problems, including the shallow water equations, to interpolate gravity at any given location using only measurements at some discrete locations, etc.

To apply Least-Squares Collocation Method, we first chose a finite set of $N = 396$ collocation points $\{X_j\}_{j=1,\ldots,396} = \{\{-90:20:110\},\{-120:20:-20\},\{2:2:12\}\}$ in the measurements domain $\Omega = [-98,118]\times[-125,3]\times[1,13]$.

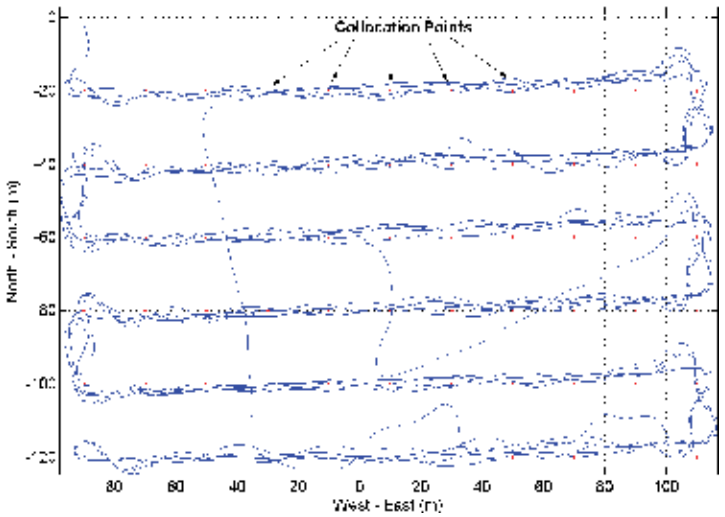


Fig. 7. Plan view of the AUV plume tracking survey and collocation points.

Then an approximation of the desired variable (in this case, salinity, temperature and density, denoted by P) between measured and collocation points in the form was assumed

$$P_i \approx \sum_{j=1}^{N} W_{ij}\hat{P}_j, \quad i = 1,\ldots n \Leftrightarrow P = W\hat{P} \tag{3}$$

where P_i is the measurement at point X_i, with $n = 16075$ observations, $\hat{P}_j$ represents the approximated measurement at collocation point X_j, and W_{ij} is an elementary function usually built in such a way that it takes a certain value if X_i is in the influence domain of point X_j, the region Ω_j, and vanishes outside the region Ω_j surrounding the point X_j. W is a n by N matrix $(n > N)$ where the column vector j represents the magnitudes of the approximated measurement of collocation point X_j with respect to each observation i, and where the row vector i represents the magnitudes of the approximated measurements of each collocation point X_j with respect to observation point X_i.

The choice for the elementary function W_{ij} was the following raised cosine function

$$W_{ij}(d) = \begin{cases} \frac{1}{2} + \frac{1}{2}\cos(\pi d), & 0 \le d < 1 \\ 0 \quad , & d \ge 1 \end{cases} \tag{4}$$

where d is a normalized distance given by

$$d\left(X_i, X_j\right) = \left\| X_i - X_j \right\|_{nrm} = \left[\left(\frac{\left| x_i - x_j \right|}{\Delta x} \right)^{nrm} + \left(\frac{\left| y_i - y_j \right|}{\Delta y} \right)^{nrm} + \left(\frac{\left| z_i - z_j \right|}{\Delta z} \right)^{nrm} \right]^{\frac{1}{nrm}} \tag{5}$$

being $nrm = 2.45$, $X_i = \left(x_i, y_i, z_i\right)$, $X_j = \left(x_j, y_j, z_j\right)$, and $\Delta x = 20$, $\Delta y = 20$ and $\Delta z = 2$ the cell grid distances, respectively, in x, y and z axis, between consecutive collocation points.

If X_i is in the influence domain of point X_j, the value W_{ij} is as large as less distanced are the points X_i and X_j, being a unit value when $X_j = X_i$ (and null for all the other collocation points).

Since the magnitudes are constants, Equation (3) is an overdetermined linear system of equations (n equations with N unknowns) that can be solved using least squares method, by one of the several mathematical packages such as *Matlab*, the one used in this case.

The least squares functional is defined by summing the squares of the residuals evaluated for each point X_i

$$J = \sum_{i=1}^{n} \left[P_i - \sum_{j=1}^{N} W_{ij} \hat{P}_j \right]^2 = \left(P - W\hat{P} \right)^T \left(P - W\hat{P} \right). \tag{6}$$

Minimization of Equation (6) with respect to $\hat{P}$ leads to

$$\left(W^T W \right) \hat{P} = W^T P \tag{7}$$

where W^T is the transpose of matrix W. Solving the resulting system (of N equations with N unknowns) for $\hat{P}$ we obtain

$$\hat{P} = \left(W^T W \right)^{-1} W^T P. \tag{8}$$

An important advantage of this method is the sparsity of the matrix W. Since W has a high percentage of zero-valued elements, using a sparse matrix data only the nonzero elements and their indices are stored, reducing significantly the amount of memory for storage, and making the matrix inversion calculation process more efficient. Only a few seconds were necessary for the matrix inversion process, being W a 16075 by 396 matrix.

Notice that d was not defined exactly as Euclidean distance, i.e. L_2 norm, but a $L_{2.45}$ norm.

The explanation for this is illustrated in Fig. 8: plots (a) and (b) show the behaviour of function W for a 1D measurement domain $\Omega = [0,4]$, and collocation points $\left\{X_j\right\}_{j=1,2,3} = \{1,2,3\}$ with $\Omega_1 = [0,2]$, $\Omega_2 = [1,3]$ and $\Omega_3 = [2,4]$; plots (c), (d), and (e) show the behaviour of function W for a 2D measurement domain $\Omega = [0,3] \times [0,3]$, and collocation points $\left\{X_j\right\}_{j=1,2,3,4} = \{(1,1),(2,1),(1,2),(2,2)\}$ with $\Omega_1 = [0,2] \times [0,2]$, $\Omega_2 = [1,3] \times [0,2]$, $\Omega_3 = [0,2] \times [1,3]$, and $\Omega_4 = [1,3] \times [1,3]$ using L_2 norm, and plots (f), (g), and (h) are the same as before except using $L_{2.45}$ norm.

As can be seen from plot (b), in the 1D measurement domain case, the sum of magnitudes of the approximated measurements of each collocation point X_j with respect to a observation

point X_i, with $d = |x_i - x_j|$ in the influence domain of point X_j is, as expected, a unit value, except in the boundaries of course. In the 2D measurement domain case, as can be seen from plot (d) and (e), the sum of magnitudes of the approximated measurements of each collocation point X_j with respect to a observation point X_i, with $d = \sqrt{|x_i - x_j|^2 + |y_i - y_j|^2}$, the Euclidean distance, in the influence domain of point X_j, is not as should, a unit value, except of course in the boundaries.

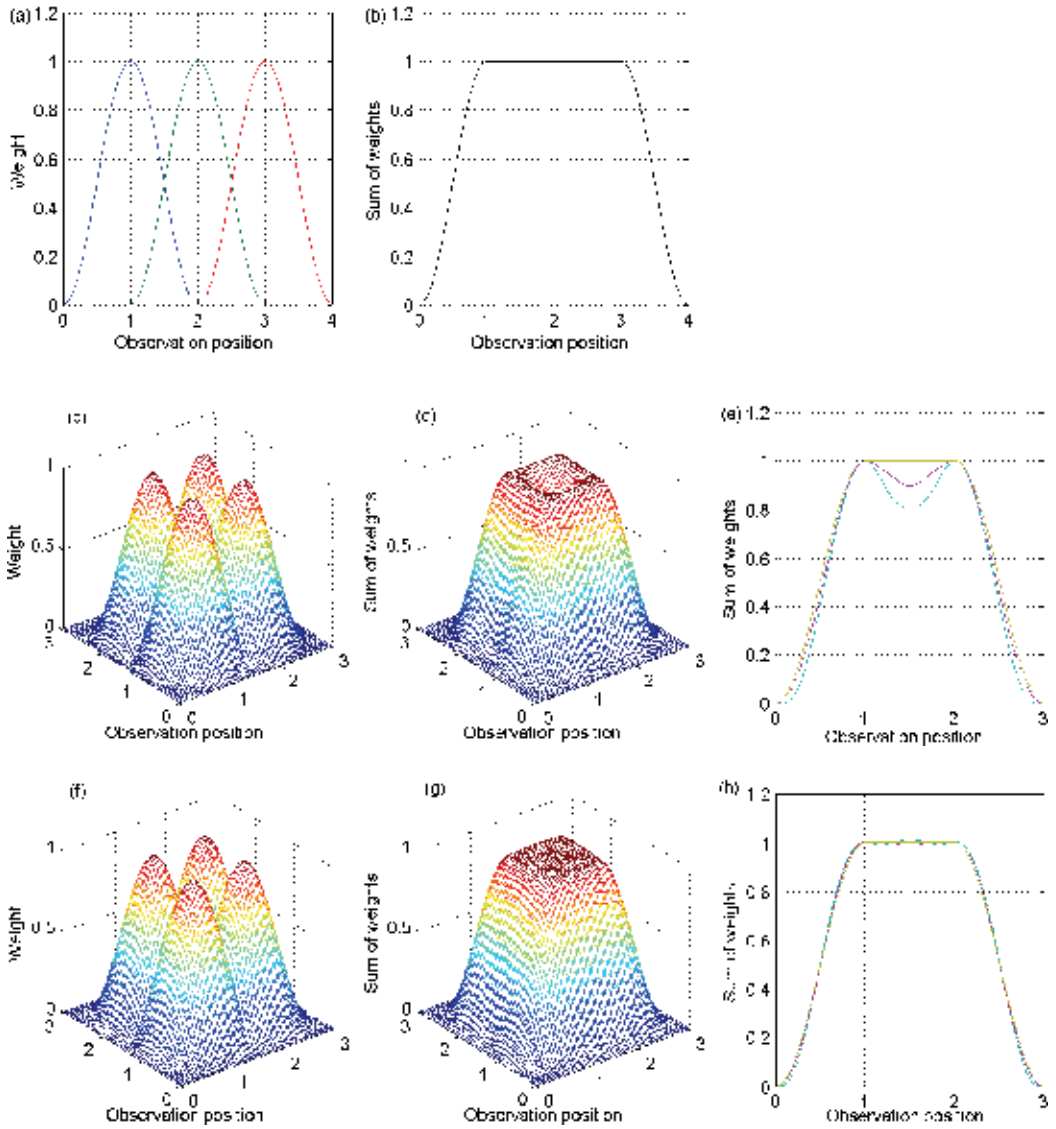


Fig. 8. Behaviour of weight function W for 1D and 2D measurement domains, using norm L_2 and $L_{2.45}$ in 2D case.

This problem is partially eliminated if a norm $L_{2.45}$ is used, as show plots (g) and (h). This value was empirically adjusted so that the sum of magnitudes of the measurements of each collocation point be approximated a unit value. Cross sections for the 3D measurements domain case (not shown) were performed and the results are similar to the 2D case previously presented.

A "less visited collocation point" X_j was defined as one whose sum of magnitudes (sum of elements of column vector j) was less then the difference between the mean value of the all sums of magnitudes $(SumMag)$ and three times the standard deviation of these sums

$$lessvisit_j = \sum_{i=1}^{n} W_{ij} < \left(mean\left(SumMag\right) - 3 \times std\left(SumMag\right) \right). \tag{9}$$

To increase information on the desired variable in the vicinity of less visited collocation points X_j, a cell grid size update was performed

$$\left[\Delta x\, \Delta y\, \Delta z\right]_j = cell\_growth \times \left[\Delta x\, \Delta y\, \Delta z\right] \tag{10}$$

where $cell\_growth = 5$.

Finally, a finer meshgrid of the form $\left[\Delta x\, \Delta y\, \Delta z\right] = \left[2\ 2\ 0.2\right]$ was considered for the surface visualization generation. The desired variable on the M visualization points was calculated as follows

$$\widetilde{P}_k = \sum_{j=1}^{N} \widetilde{W}_{kj} \hat{P}_j + mean\left(P\right), \quad k = 1, \ldots, M \tag{11}$$

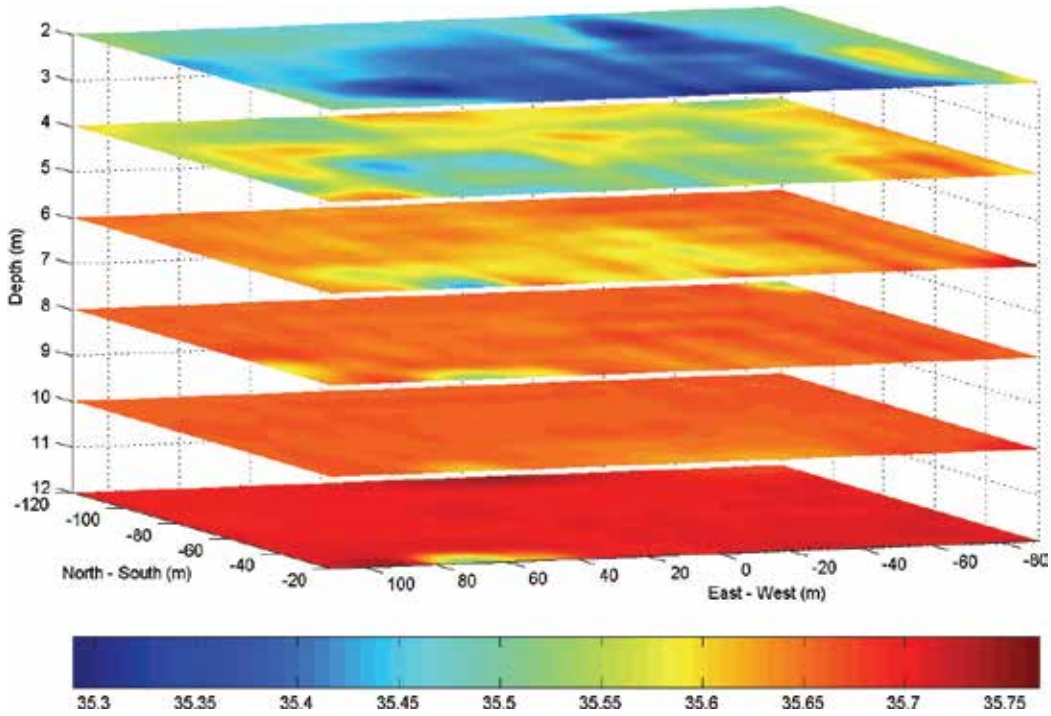
were the weights matrix $\widetilde{W}_{kj}$ was evaluated in the same manner as (4). To improve the method efficiency, the least squares solution (8) was either computed as follows

$$\hat{P} = \left(W^T W\right)^{-1} WT\left(P - mean\left(P\right)\right) \tag{12}$$

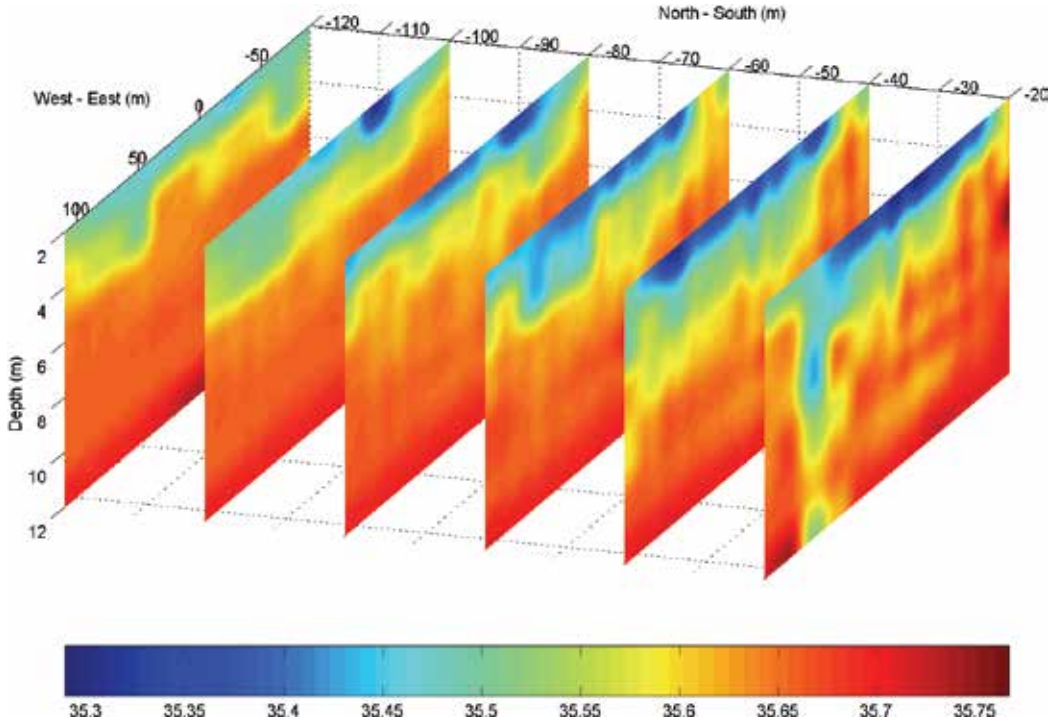
so the mean value has to be added in (11). Several interpolation methods such as Nearest Neighbor, Bilinear, and Bicubic were first applied to the measured data but with no successful results. The LSCM was then considered since it is specially attractive for problems posed on irregularity shaped domains, which is the case here. Giving the intermittence of the phenomena in observation, using "local" functions instead of using elementary functions which cover the all measurements domain (e.g., Fourier Series), no influence is assumed between widely estimated measurements of the desired variable.

3.1.2 Results

As in other field studies (Washburn et al., 1992; Petrenko et al. 1998; Jones et al. 2001), salinity was found to be more useful than either temperature or density in delineating and observing the plume structure. The LSCM results for the salinity parameter are presented in Fig. 9. From those figures, plotted with the same color scale, it is possible to identify unambiguously the effluent plume and to observe its dispersion downstream in the North-South direction. It appears as a region of lower salinity compared to surrounding ocean


Depth (m)
2
3
4
5
6
7
8
9
10
11
12
North - South (m)
-120
-100
-80
-60
-40
-20
East - West (m)
100
80
60
40
20
0
-20
-40
-60
35.3
35.35
35.4
35.45
35.5
35.55
35.6
35.65
35.7
35.75

(**a**)


North - South (m)
-120
-110
-100
-90
-80
-70
-60
-50
-40
-30
West - East (m)
-50
0
50
100
Depth (m)
2
4
6
8
10
12
35.3
35.35
35.4
35.45
35.5
35.55
35.6
35.65
35.7
35.75

(**b**)

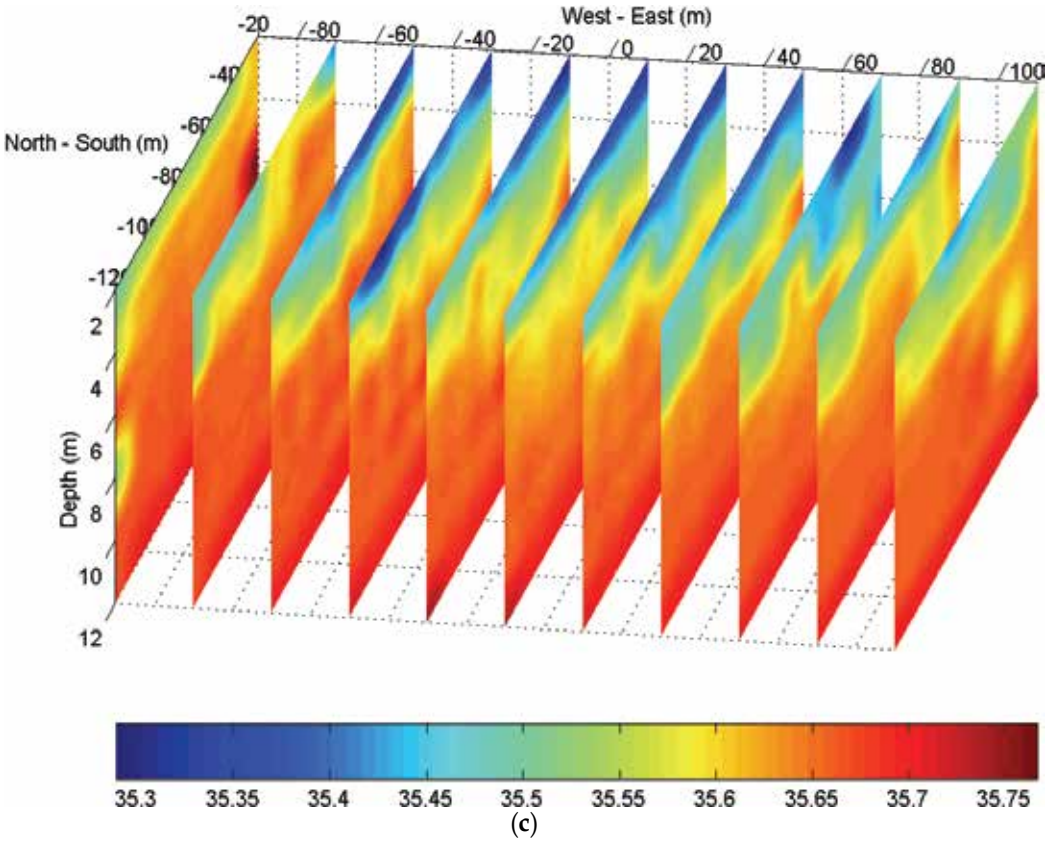


Fig. 9. (a) Salinity transversal sections (psu units) at 20, 40, 60, 80, 100 and 120 m downstream from the middle point of the diffuser; (b) Salinity horizontal sections (psu units) at 2, 4, 6, 8, 10, and 12 m depth; Salinity longitudinal sections (psu units) from –20 m West to 100 m East.

waters at the same depth, rising to the water surface due to the relatively weak stratification, low currents and shallow waters.

In the 20 m transversal section and on the several horizontal sections in the region close to the diffuser it is possible to observe the plume rising from near the bottom to the surface, in accordance with Fig. 5 (see that the East end of the diffuser is over the first transect). South from the diffuser, downstream, there is also evidence of the presence of the effluent plume at the surface, with salinity decreasing to the edges.

Major differences in salinity between the plume and surrounding waters at the surface was observed to be about 0.4 psu in the first two sections, decreasing to about 0.15 psu in the third and fourth sections, and being less than 0.1 psu in the fifth section, finally being almost equal to that of background waters at 120 m distance from the diffuser.

Salinity anomalies of the same order were found by Washburn et al. (1992) and Petrenko et al. (1998). Vertical profiles of salinity collected by Petrenko et al. (1998) at the center and over the western end of the diffuser, where the highest effluent concentrations were found, indicated differences of 0.2 psu. Typical salinity anomalies in the plume of the order of 0.1 psu were observed by Washburn et al. (1992).

The effluent plume was detected from close to the surface (at minimum depths around 1.5 m) to nearly 8 m depth from the first to the fourth section with clearly decreasing thickness downstream. A sharp difference in salinity at the effluent plume lateral edges is clearly visible.

The form of the wastefield spreading (almost centered in the survey area) indicates that the sampling strategy designed was very successful, even for a surfacing plume. A surfacing plume, several times more diluted than a submerged plume, surrounded by low salinity surface waters, with its own weak signals, could be further blurred by the background signals (Petrenko et al., 1998).

The plume exhibits a considerably more complex structure than the compact shape of the classical picture of the buoyant plume but is not so patchy as in previous studies, perhaps because of improvements in horizontal and vertical resolution (Faisst et al., 1990; Petrenko et al., 1998; Jones et al., 2001; Carvalho et al., 2002).

Roberts et al. (1989) laboratory experiments on multiport diffusers in density-stratified perpendicular crossflows show that at low current speeds $(F \approx 0.1)$ the flow has the normal plume-like pattern with the plume bent downstream. At higher current speeds $(F \approx 10)$ the plume cannot entrain all of the incoming flow while maintaining the free plume pattern and the base of the wastefield stays at the nozzle level. This is known as the forced entrainment regime, which occurs when the Froude number F defined above exceeds a value which lies somewhere between 1 and 10. The rise height and thickness of the wastefield decrease with increasing current speed in the forced entrainment regime. Our *in situ* observations and measured value of the Froude number seem to be in agreement with these experiments. The observations indicate that the plume is being swept downstream and not attached to the lower boundary, a regime that lies between the two mentioned, in agreement with the behavior expected for a measured Froude number $F = 1.18$.

3.2 Dilution estimation

Dilution was estimated empirically using temperature and salinity and their representation on a TS – diagram, with initial mixing lines between sewage effluent and receiving waters. Details of this method may be found in Washburn et al. (1992) and Petrenko et al. (1998).

When sewage effluent (with temperature T_e and salinity S_e is discharged, it starts mixing with receiving waters (with temperature T_a and salinity S_a) at the port level. The temperature and salinity of the mixed water mass, respectively T_m and S_m, correspond to a point of the mixing line connecting the effluent and the ambient TS – points. The characteristics T_m and S_m vary according to the dilution factor between the effluent and the receiving waters. For a given dilution S, T_m and S_m are equal to (Fischer et al. 1979)

$$\begin{cases} T_m = T_a + \dfrac{T_e - T_a}{S} \\ S_m = S_a + \dfrac{S_e - S_a}{S} \end{cases} \tag{13}$$

As previously mentioned, mixing in the near-field occurs during the plume rise over the outfall diffuser. The observations indicate that some of this initial mixing occurs between 15 and 11 m depth. Two ambient TS – points from those two depths were then considered to account for variability in background conditions.

Fig. 10 shows temperature and salinity measured at Section 1 plotted in a $TS-$ diagram. $TS-$ points measured at the same depth were plotted with the same color: red, blue and black correspond respectively to 12, 10 and 8~m depth. The isopycnals (lines of constant density) are labelled in sigma units.

Simultaneously, initial mixing lines are drawn between the effluent $TS-$ point (not shown) $T_e = 35^o$ and $S_e = 2$ psu, and the two ambient $TS-$ points (T_a and S_a) at 15 and 11 m depth. $TS-$ points of these two mixing lines with equal dilutions delimit intervals of initial dilutions. $TS-$ points (T_m and S_m) for dilution factors of 30, 35, 40, 45, 50, 55, 60, 70, 80, 90, 100, 120, 150, 200, 300 and 400 are indicated.

Dilution can be estimated empirically based on the fact that $TS-$ points in the plume are single end-members of the mixing between the effluent and the receiving waters. The $TS-$ points in the plume with the lowest dilution falling into an initial dilution zone establish the minimum initial dilution.

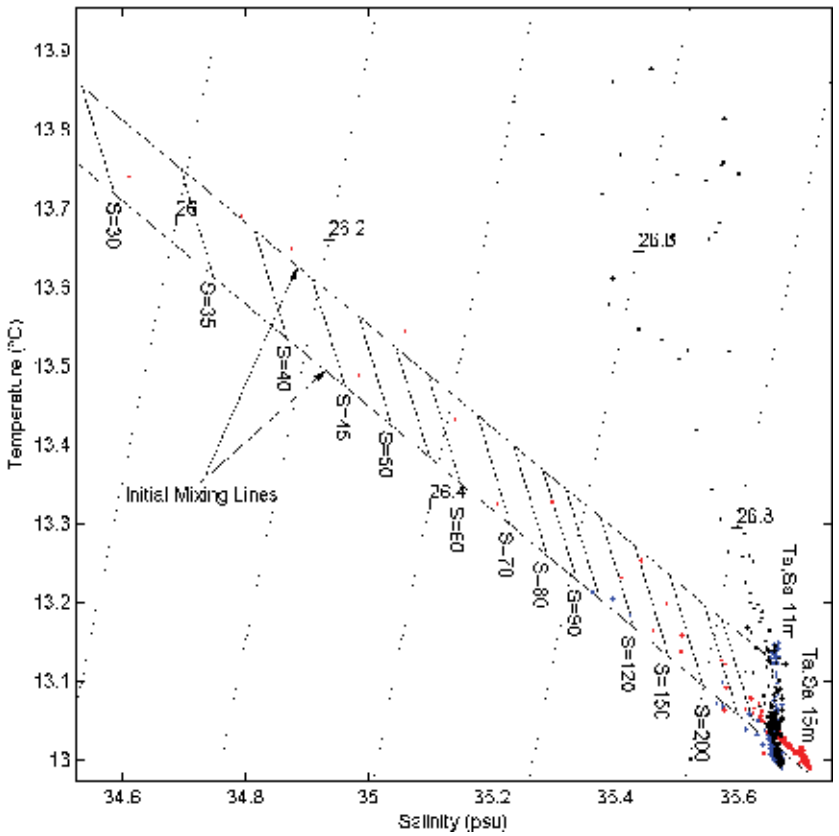


Fig. 10. Temperature-Salinity $(TS-)$ diagram of data from Section 1.

The $TS-$ points in the plume located off the initial mixing lines have already started to mix with waters above 11 m depth. According to these results, effluent dilutions were then at least 30:1. This value is probably a lower bound of the dilution since, in reality, mixing continued to occur up to surface. However, with this method no further dilution can be inferred.

4. Conclusion

An oceanographic campaign was performed on July 30, 2002 to study the shape and dilution of the *S. Jacinto* outfall plume using *Isurus* AUV.

Our results demonstrate that AUVs can provide high-quality measurements of physical (and probably optical) properties of effluent plumes in a quite effective manner. An efficient sampling strategy, enabling improvements in terms of resolution of time and space scales and undersampling, demonstrated that effluent plumes can be clearly traced using naturally-occurring tracers in the wastewater.

In order to reduce the uncertainty about plume location and to concentrate the vehicle mission only in the hydrodynamic mixing zone, outputs of a near-field prediction model, based on *in situ* measurements of current speed and direction and density stratification obtained in real-time, were used to specify the AUV mission.

A built-in application adaptively specified the AUV monitoring field transects according to the environment conditions, in real-time, taking into account the outputs of the model and the vehicle navigation requirements.

A data processing system was created, applying the Least Squares Collocation Method (LSCM) technique, in order to map effectively the dispersion of the effluent using the AUV data. LSCM results for salinity enable the effluent plume to be identified unambiguously and its dispersion downstream to be observed. The effluent plume appeared as a region of lower salinity compared to surrounding ocean waters at the same depth, rising to the water surface due to the relatively weak ambient stratification and relatively weak low currents.

Dilution was estimated using temperature-salinity, by means of a $(TS-)$ diagram. The analysis demonstrated that effluent dilutions were at least 30:1 in this study. Dilutions estimated with the $TS-$ method represent lower bounds of dilution, specially for surfacing plumes.

If artificial tracers had been used, better estimates of dilution would certainly have been obtained. However, since the present monitoring methodology is considerably less expensive and more practical for routine monitoring (not forgetting the negative impacts of releasing fluorescent dyes or other contaminant components in the effluent) further developments with the present system will certainly be justifiable.

Spectral fluorescence methods provide some promise for observing potentially-unique characteristics of effluent plumes (Petrenko et al., 1997). In the near future it would be interesting to *Isurus* AUV to test the potential of fluorescence measurements as a non-invasive, real-time technique to detect sewage fields in the coastal environment. AUVs also appear to be quite promising for studying the patchiness problem, in spite of several limitations that must be overcome in the future, such as predicting variability over broad time and space scales.

5. References

Alt, C.V.; Allen, B.; Austin, T. & Stokey, R. (1994). Remote Environmental Measuring Units, *Proceedings of the Autonomous Underwater Vehicles '94 Conference*, July 1994.

Carvalho, J.L.B.; Roberts, P.J.W. & Roldão, J. (2002). Field Observations of the Ipanema Beach Outfall, *Journal of Hydraulic Engineering*, Vol. 128, No. 2, pp. 151-160.

Faisst, W.K.; McDonald, R.M.; Noon, T. & Marsh, G. (1990). Iona Outfall, Plume Characterization Study, *Proceedings 1990 National Conference on Hydraulic Engineering*, ASCE, July 30 - August 3, 1990.

Fischer, H.B.; List, J.E.; Koh, R.C.Y.; Imberger, J. & Brooks, N.H. (1979). *Mixing in Inland and Coastal Waters*, Academic Press.

Fletcher, B. (2001). Chemical Plume Mapping with an Autonomous Underwater Vehicle, *Proceedings of MTS/IEEE International Conference Oceans 2001*, Biloxi, Hawaii, USA, November 5-8, 2001, pp. 508-512.

Jones, B.H.; Barnett, A. & Robertson, G.L. (2001). Towed Mapping of the Effluent Plume from a Coastal Ocean Outfall, *Proceedings of MTS/IEEE International Conference Oceans 2001*, MTS 0-933957-29-7, Biloxi, Hawaii, USA, November 5-8, 2001, pp. 1985-1989.

Matos, A.; Cruz, N.; Martins, A. & Pereira, F. L. (1999). Development and Implementation of a Low-Cost LBL Navigation System for an AUV, *Proceedings of the MTS/IEEE Oceans'99 Conference*.

Matos, A.; Cruz, N. & Pereira, F.L. (2003). Post Mission Trajectory Smoothing for the Isurus AUV, *Proceedings of Oceans 2003 Marine Technology and Ocean Science Conference*, September, 2003.

Millero, F.J.; Chen, C.T.; Bradshaw, A. & Schleicher K. (1980). A New High Pressure Equation of State for Seawater, *Deep-Sea Research 27A*, pp. 255-264.

Petrenko, A.A.; Jones, B.H.; Dickey, T.D.; LeHaitre, M. & Moore, C. (1997). Effects of a Sewage Plume on the Biology, Optical Characteristics, and Particle Size Distributions of Coastal Waters. Journal of Geophysical Research, Vol. 102, No. C11, pp. 25061-25071.

Petrenko, A.A.; Jones, B.H. & Dickey, T.D. (1998). Shape and Initial Dilution of Sand Island, Hawaii Sewage Plume, *Journal of Hydraulic Engineering*, Vol. 124, No. 6, pp. 565-571.

Ramos, P. (2005). Advanced Mathematical Modeling for Outfall Plume Tracking and Management using Autonomous Underwater Vehicles based Systems, *PhD Thesis*, Faculty of Engineer of University of Porto, March 2005.

Roberts, P.J.W.; Snyder, W. & Baumgartner, D. (1989). Ocean Outfalls, *Journal of Hydraulic Engineering*, Vol. 115, No. 1, pp. 1-70.

Roberts, P.J.W. & Wilson, D. (1990). Field and Model Studies of Ocean Outfalls, *Hydraulic Engineering Proceedings, 1990 National Conference*, ASCE, H. Chang, New York, San Diego, July 30 - August 3, 1990.

Roberts, P.J.W. (1996). Sea Outfalls, *Environmental Hydraulics*, V. P. Singh and W. H. Hager, Kluwer Academic Press, pp. 63-110.

Roberts, P.J.W.; Hunt, C.D. & Mickelson, M.J. (2002). Field and Model Studies of the Boston Outfall, *Proceedings of the 2nd International Conference on Marine Waste Water Discharges*, Istanbul, Turkey, September 16-21, 2002.

Robinson, A.R.; Bellingham, J.G.; Chryssostomidis, C.; Dickey, T.D.; Levine, E.; Petrikalakis, N.; Porter, D.L.; Rothschild, B.J.; Schmidt, H.; Sherman, K.; Holliday, D.V. & Atwood, D.K. (1999). Real-Time Forecasting of the Multidisciplinary Coastal Ocean with the Littoral Ocean Observing and Predicting System (LOOPS), *Proceedings of the Third Conference on Coastal Atmospheric and Oceanic Prediction Processes*, American Meteorological Society, New Orleans, LA.

Washburn, L.; Jones, B.H.; Bratkovich, A.; Dickey, T.D. & Chen, M. (1992). Mixing, Dispersion, and Resuspension in Vicinity of Ocean Wastewater Plume, *Journal of Hydraulic Engineering*, Vol. 118, No. 1, pp. 38-58.

Wu, Y.; Washburn, L. & Jones, B.H. (1994). Buoyant Plume Dispersion in a Coastal Environment: Evolving Plume Structure and Dynamics, *Continental Shelf Research*, Vol. 14, No. 9, pp. 1001-1023.

Yu, X.; Dickey, T.D.; Bellingham, J.G.; Manov, D. & Streitlien, K. (1994). The Application of Autonomous Underwater Vehicles for Interdisciplinary Measurements in Massachusetts and Cape Cod Bayes, *Continental Shelf Research*, Vol. 22, No. 15, pp. 2225-2245.

Zhang, X.; Liu, X.; Song, K. & Lu, M. (2001). Least-Squares Collocation Meshless Method, *Int. Journal for Numerical Methods in Engineering*, Vol. 51, pp. 1089-1100.

23

Resolved Acceleration Control for Underwater Vehicle-Manipulator Systems: Continuous and Discrete Time Approach

Shinichi Sagara
Kyushu Institute of Technology
Japan

1. Introduction

Underwater robots, especially Underwater Vehicle-Manipulator Systems (UVMS), are expected to have important roles in ocean exploration (Yuh, 1995). Many studies about dynamics and control of UVMS have been reported (Maheshi et al., 1991; McMillan et al., 1995; McLain et al., 1996; Tarn et al., 1996; Antonelli & Chiaverini, 1998; McLain et al., 1998; Antonelli et al., 2000; Sarkar & Podder, 2001). However, there are only a few experimental studies. Most of the control methods of UVMS have been proposed based on the methods of Autonomous Underwater Vehicles. In these control methods, the desired accelerations and velocities of the end-tip of the manipulator are transformed to the desired manipulator's joint accelerations and velocities only use of the kinematic relation, and the computed torque method with joint angle and angular velocity feedbacks are utilized. In other words, the control methods use errors consisting of task-space signals of vehicle and joint-space signals of manipulator. Therefore, the control performance of the end-effector depends on the vehicle's control performance.

We have proposed continuous-time and discrete-time Resolved Acceleration Control (RAC) methods for UVMS (Yamada & Sagara, 2002; Sagara, 2003; Sagara et al., 2004; Sagara et al., 2006; Yatoh & Sagara, 2007; Yatoh & Sagara, 2008). In our proposed methods, the desired joint values are obtained by kinematic and momentum equations with feedback of task-space signals. From the viewpoint of underwater robot control, parameters and coefficients of hydrodynamic models are generally used as constant values that depend on the shape of the robots (Fossen, 1994). Our proposed methods described above can reduce the influence of the modelling errors of hydrodynamics by position and velocity feedbacks. The effectiveness of the RAC methods has been demonstrated by using a floating underwater robot with vertical planar 2-link manipulator shown in Figure 1.

In this chapter, our proposed continuous-time and discrete-time RAC methods are described and the both experimental results using a 2-link underwater robot are shown. First, we explain about a continuous-time RAC method and show that the RAC method has good control performance in comparison with a computed torque method. Next, to obtain higher control performance, we introduce a continuous-time RAC method with disturbance compensation. In practical systems digital computers are utilized for controllers, but there is no discrete-time control method for UVMS except our proposed methods. Then, we address

discrete time RAC methods including the ways of disturbance compensation and avoiding singular configuration.

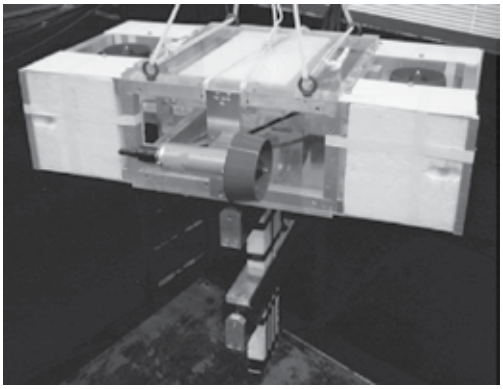


Fig. 1. Vertical type 2-link underwater robot

2. Modelling

The UVMS model used in this chapter is shown in Figure 2. It has a robot base (vehicle) and an n-DOF manipulator.

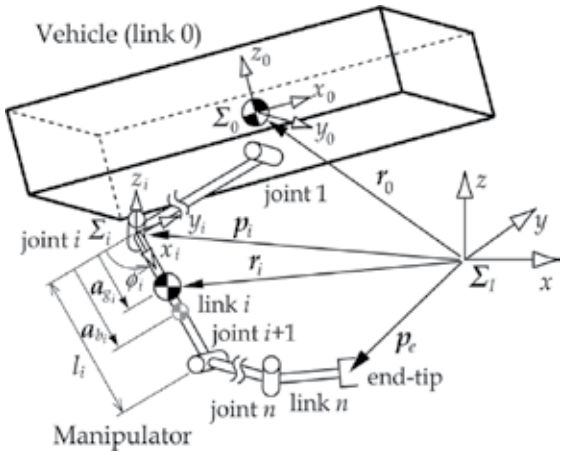


Fig. 2. Model of underwater robot with n-link manipulator

The symbols used in this chapter are defined as follows:

n : number of joints

Σ_I : inertial coordinate frame

Σ_i : link i coordinate frame (i = 0, 1, 2, $\cdots$, n; link 0 means the vehicle)

${}^I\boldsymbol{R}_i$: coordinate transformation matrix from Σ_i to Σ_I

$\boldsymbol{p}_e$: position vector of the end-tip of the manipulator with respect to Σ_I

$\boldsymbol{p}_i$: position vector of the origin of Σ_i with respect to Σ_I

$\boldsymbol{r}_i$: position vector of the center of gravity of link i with respect to Σ_I

ϕ_i : relative angle of joint i

$\boldsymbol{\psi}_0$: roll-pitch-yaw attitude vector of Σ_0 with respect to Σ_I

$\boldsymbol{\psi}_e$: roll-pitch-yaw attitude vector of the end-tip of the manipulator with respect to Σ_I

$\boldsymbol{\omega}_0$: angular velocity vector of Σ_0 with respect to Σ_I

$\boldsymbol{\omega}_e$: angular velocity vector of the end-tip of the manipulator with respect to Σ_I

$\boldsymbol{\phi}$: relative joint angle vector ($=[\phi_1 \quad \cdots \quad \phi_n]^T$)

${}^i\boldsymbol{k}_i$: unit vector indicating a rotational axis of joint i ($=[0 \quad 0 \quad 1]^T$)

m_i: mass of link i

${}^i\boldsymbol{M}_{a_i}$: added mass matrix of link i with respect to Σ_i

${}^i\boldsymbol{I}_i$: inertia tensor of link i with respect to Σ_i

${}^i\boldsymbol{I}_{a_i}$: added inertia tensor of link i with respect to Σ_i

$\boldsymbol{x}_0$: position and orientation vector of Σ_0 with respect to Σ_I ($=[\boldsymbol{r}_0^T \quad \boldsymbol{\Psi}_0^T]^T$)

$\boldsymbol{x}_e$: position and orientation vector of the end-tip with respect to Σ_I ($=[\boldsymbol{p}_e^T \quad \boldsymbol{\Psi}_e^T]^T$)

$\boldsymbol{\nu}_0$: linear and angular vector of Σ_0 with respect to Σ_I ($=[\dot{\boldsymbol{r}}_0^T \quad \boldsymbol{\omega}_0^T]^T$)

$\boldsymbol{\nu}_e$: linear and angular vector of the end-tip with respect to Σ_I ($=[\dot{\boldsymbol{p}}_e^T \quad \boldsymbol{\omega}_e^T]^T$)

l_i: length of link i

$\boldsymbol{a}_{g_i}$: position vector from joint i to the center of gravity of link i with respect to Σ_I

$\boldsymbol{a}_{b_i}$: position vector from joint i to the buoyancy center of link i with respect to Σ_I

D_i : width of link i

V_i : volume of link i

ρ : fluid density

C_{D_i} : drag coefficient of link i

$\boldsymbol{g}$: gravitational acceleration vector

$\boldsymbol{E}_j$: $j \times j$ unit matrix

$\tilde{\cdot}$: tilde operator stands for a cross product such that $\tilde{\boldsymbol{r}}\boldsymbol{a} = \boldsymbol{r} \times \boldsymbol{a}$

2.1 Kinematics

First, from Figure 2 a time derivative of the end-tip position vector $\dot{\boldsymbol{p}}_e$ is

$$\dot{\boldsymbol{p}}_e = \dot{\boldsymbol{r}}_0 + \tilde{\boldsymbol{\omega}}_0(\boldsymbol{p}_e - \boldsymbol{r}_0) + \sum_{i=1}^{n} \left\{ \tilde{\boldsymbol{k}}_i(\boldsymbol{p}_e - \boldsymbol{p}_i) \right\} \dot{\phi}_i \tag{1}$$

where $\boldsymbol{k}_i = {}^I\boldsymbol{R}_i \, {}^i\boldsymbol{k}_i$.

On the other hand, relationship between end-tip angular velocity and joint velocity is expressed with

$$\boldsymbol{\omega}_e = \boldsymbol{\omega}_0 + \sum_{i=1}^{n} \boldsymbol{k}_i \dot{\phi}_i \, . \tag{2}$$

From Equations (1) and (2) the following equation is obtained:

$$\boldsymbol{\nu}_e = \boldsymbol{A}\boldsymbol{\nu}_0 + \boldsymbol{B}\dot{\boldsymbol{\phi}} \tag{3}$$

where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{E}_3 & -(\tilde{\boldsymbol{p}}_e - \tilde{\boldsymbol{r}}_0) \\ \boldsymbol{0} & \boldsymbol{E}_3 \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} \tilde{\boldsymbol{k}}_1(\boldsymbol{p}_e - \boldsymbol{p}_1) & \tilde{\boldsymbol{k}}_2(\boldsymbol{p}_e - \boldsymbol{p}_2) & \cdots & \tilde{\boldsymbol{k}}_n(\boldsymbol{p}_e - \boldsymbol{p}_n) \\ \boldsymbol{k}_1 & \boldsymbol{k}_2 & \cdots & \boldsymbol{k}_n \end{bmatrix}.$$

Next, let $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ be a linear and an angular momentum of the robot including hydrodynamic added mass tensor ${}^i\boldsymbol{M}_{a_i}$ and added inertia tensor ${}^i\boldsymbol{I}_{a_i}$ of link *i*. Then

$$\boldsymbol{\eta} = \sum_{i=0}^{n} \boldsymbol{M}_{T_i}\dot{\boldsymbol{r}}_i\,, \tag{4}$$

$$\boldsymbol{\mu} = \sum_{i=0}^{n} \left(\boldsymbol{I}_{T_i}\boldsymbol{\omega}_i + \tilde{\boldsymbol{r}}_i\boldsymbol{M}_{T_i}\dot{\boldsymbol{r}}_i\right) - \boldsymbol{r}_0 \times \boldsymbol{\eta} \tag{5}$$

where $\boldsymbol{M}_{T_i} = m_i\boldsymbol{E}_3 + {}^I\boldsymbol{R}_i\ {}^i\boldsymbol{M}_{a_i}\ {}^i\boldsymbol{R}_I$ and $\boldsymbol{I}_{T_i} = {}^I\boldsymbol{R}_i\,({}^i\boldsymbol{I}_i + {}^i\boldsymbol{I}_{a_i})\ {}^i\boldsymbol{R}_I$. Here, linear and angular velocities of the center of gravity of link *i* are described as

$$\dot{\boldsymbol{r}}_i = \dot{\boldsymbol{r}}_0 + \tilde{\boldsymbol{\omega}}_0(\boldsymbol{r}_i - \boldsymbol{r}_0) + \sum_{j=1}^{i} \left\{\tilde{\boldsymbol{k}}_j(\boldsymbol{r}_i - \boldsymbol{p}_j)\right\}\dot{\phi}_j\,, \tag{6}$$

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_0 + \sum_{j=1}^{i} \boldsymbol{k}_j\dot{\phi}_j\,. \tag{7}$$

Therefore, the following equation is obtained from Equations (4)-(7):

$$\boldsymbol{s} = \begin{bmatrix} \boldsymbol{\eta} \\ \boldsymbol{\mu} \end{bmatrix} = \boldsymbol{C}\boldsymbol{\nu}_0 + \boldsymbol{D}\dot{\boldsymbol{\phi}} \tag{8}$$

where

$$\boldsymbol{C} = \begin{bmatrix} \sum_{i=0}^{n}\boldsymbol{M}_{T_i} & -\sum_{i=0}^{n}\boldsymbol{M}_{T_i}(\tilde{\boldsymbol{r}}_i - \tilde{\boldsymbol{r}}_0) \\ \sum_{i=0}^{n}(\tilde{\boldsymbol{r}}_i - \tilde{\boldsymbol{r}}_0)\boldsymbol{M}_{T_i} & \sum_{i=0}^{n}\left\{\boldsymbol{I}_{T_i} - (\tilde{\boldsymbol{r}}_i - \tilde{\boldsymbol{r}}_0)\boldsymbol{M}_{T_i}(\tilde{\boldsymbol{r}}_i - \tilde{\boldsymbol{r}}_0)\right\} \end{bmatrix},$$

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{d}_{11} & \boldsymbol{d}_{12} & \cdots & \boldsymbol{d}_{1n} \\ \boldsymbol{d}_{21} & \boldsymbol{d}_{22} & \cdots & \boldsymbol{d}_{2n} \end{bmatrix},$$

$$\boldsymbol{d}_{1i} = \sum_{j=i}^{n} \boldsymbol{M}_{T_j} \tilde{\boldsymbol{k}}_i \left(\boldsymbol{r}_j - \boldsymbol{p}_i \right),$$

$$\boldsymbol{d}_{2i} = \sum_{j=i}^{n} \left\{ \boldsymbol{I}_{T_j} \boldsymbol{k}_i + \left(\tilde{\boldsymbol{r}}_j - \tilde{\boldsymbol{r}}_0 \right) \boldsymbol{M}_{Tj} \tilde{\boldsymbol{k}}_i \left(\boldsymbol{r}_j - \boldsymbol{p}_i \right) \right\}.$$

Here, we assume that the added mass and added inertia are constant. In reality, the added mass and inertia are variable, but the influence of the variation is compensated by a control method given in the following section.

2.2 Equation of motion

First, the drag force and the moment of joint *i* can generally by represented as follows (Levesque & Richard, 1994):

$$\boldsymbol{f}_{d_i} = \frac{\rho}{2} C_{D_i} D_i {}^I \boldsymbol{R}_i \int_0^{l_i} \| \boldsymbol{w}_i \| \boldsymbol{w}_i dx_i \,, \tag{9}$$

$$\boldsymbol{t}_{d_i} = \frac{\rho}{2} C_{D_i} D_i {}^I \boldsymbol{R}_i \int_0^{l_i} \hat{\boldsymbol{x}}_i \times \| \boldsymbol{w}_i \| \boldsymbol{w}_i dx_i \tag{10}$$

where

$$\boldsymbol{w}_i = \begin{bmatrix} 0 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{E}_2 \end{bmatrix} {}^i \boldsymbol{R}_I \left(\dot{\boldsymbol{r}}_i + \tilde{\boldsymbol{\omega}}_i \hat{\boldsymbol{x}}_i \right)$$

and $\hat{\boldsymbol{x}}_i = \begin{bmatrix} x_i & 0 & 0 \end{bmatrix}^T$.

Next, the gravitational and buoyant forces acting on link *i* are described as

$$\boldsymbol{f}_{g_i} = (\rho V_i - m_i) \boldsymbol{g} \,, \tag{11}$$

$$\boldsymbol{t}_{g_i} = (\rho V_i \tilde{\boldsymbol{a}}_{b_i} - m_i \tilde{\boldsymbol{a}}_{g_i}) \boldsymbol{g} \,. \tag{12}$$

Considering the hydrodynamic forces described above and using the Newton-Euler formulation, the following equation of motion can be obtained (Antonelli, 2003):

$$\boldsymbol{M}(\boldsymbol{q}) \dot{\boldsymbol{\zeta}} + \boldsymbol{N}(\boldsymbol{q}, \boldsymbol{\zeta}) \boldsymbol{\zeta} + \boldsymbol{f} = \boldsymbol{u} \tag{13}$$

where

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{\phi} \end{bmatrix}, \ \boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{v}_0 \\ \dot{\boldsymbol{\phi}} \end{bmatrix}, \ \boldsymbol{u} = \begin{bmatrix} \boldsymbol{f}_{\mathrm{B}} \\ \boldsymbol{\tau}_{\mathrm{B}} \\ \boldsymbol{\tau}_{\mathrm{M}} \end{bmatrix},$$

and $\boldsymbol{M}$ is the inertia matrix including the added mass ${}^i \boldsymbol{M}_{a_i}$ and inertia ${}^i \boldsymbol{I}_{a_i}$, $\boldsymbol{N\zeta}$ is the vector of the Coliolis and centrifugal forces, $\boldsymbol{f}$ is the vector consisting of the drag and gravitational and buoyant forces and moments, $\boldsymbol{f}_{\mathrm{B}}$ and $\boldsymbol{\tau}_{\mathrm{B}}$ are the force and torque vectors of the vehicle, respectively, and $\boldsymbol{\tau}_{\mathrm{M}}$ is the joint torque vector of manipulator. Moreover, the relationship between $\boldsymbol{\omega}_*$ and $\boldsymbol{\psi}_* = [\psi_{r_*} \quad \psi_{p_*} \quad \psi_{y_*}]^T \ (* = 0, e)$ is described as

$$\boldsymbol{\omega}_* = S_{\psi_*}\dot{\boldsymbol{\psi}}_* \tag{14}$$

where

$$S_{\psi_*} = \begin{bmatrix} \cos\psi_{p_*}\cos\psi_{y_*} & -\sin\psi_{y_*} & 0 \\ \cos\psi_{p_*}\sin\psi_{y_*} & \cos\psi_{y_*} & 0 \\ \sin\psi_{p_*} & 0 & 1 \end{bmatrix}.$$

Thus, the relationship between $\dot{\boldsymbol{q}}$ and $\boldsymbol{\zeta}$ is described as

$$\boldsymbol{\zeta} = \begin{bmatrix} E_3 & & \mathbf{0} \\ & S_{\psi_0} & \\ \mathbf{0} & & E_n \end{bmatrix}\dot{\boldsymbol{q}}\,. \tag{15}$$

3. Continuous-time RAC

3.1 RAC law

Differentiating Equations (3) and (8) with respect to time, the following equation can be obtained:

$$W(t)\boldsymbol{\alpha}(t) = \boldsymbol{\beta}(t) + \boldsymbol{\gamma}(t) - \dot{W}(t)\boldsymbol{\zeta}(t) \tag{16}$$

where

$$W = \begin{bmatrix} C + E_6 & D \\ A & B \end{bmatrix},\ \boldsymbol{\alpha} = \dot{\boldsymbol{\zeta}},\ \boldsymbol{\beta} = \begin{bmatrix} \dot{\boldsymbol{\nu}}_0 \\ \dot{\boldsymbol{\nu}}_e \end{bmatrix},\ \boldsymbol{\gamma} = \begin{bmatrix} \dot{\boldsymbol{s}} \\ \mathbf{0} \end{bmatrix},$$

and $\dot{\boldsymbol{s}}$ is the external force, including the hydrodynamic force and the thrust of the thruster, which acts on the vehicle.

For Equation (16), the reference acceleration is defined as

$$\boldsymbol{\alpha}^{\mathrm{ref}}(t) = \begin{bmatrix} \dot{\boldsymbol{\nu}}_0^{\mathrm{ref}} \\ \ddot{\boldsymbol{\phi}}^{\mathrm{ref}} \end{bmatrix} = W^{\#}(t)\{\boldsymbol{\beta}^{\mathrm{ref}}(t) + \boldsymbol{\gamma}(t) - \dot{W}(t)\boldsymbol{\zeta}(t)\} \tag{17}$$

where

$$\boldsymbol{\beta}^{\mathrm{ref}} = \begin{bmatrix} \ddot{\boldsymbol{r}}_0^{\mathrm{des}} \\ \dot{\boldsymbol{\omega}}_0^{\mathrm{des}} \\ \ddot{\boldsymbol{p}}_e^{\mathrm{des}} \\ \dot{\boldsymbol{\omega}}_e^{\mathrm{des}} \end{bmatrix} + K_V \begin{bmatrix} \dot{\boldsymbol{r}}_0^{\mathrm{des}} - \dot{\boldsymbol{r}}_0 \\ \boldsymbol{\omega}_0^{\mathrm{des}} - \boldsymbol{\omega}_0 \\ \dot{\boldsymbol{p}}_e^{\mathrm{des}} - \dot{\boldsymbol{p}}_e \\ \boldsymbol{\omega}_e^{\mathrm{des}} - \boldsymbol{\omega}_e \end{bmatrix} + K_P \begin{bmatrix} \boldsymbol{r}_0^{\mathrm{des}} - \boldsymbol{r}_0 \\ \boldsymbol{e}_{\omega_0} \\ \boldsymbol{p}_e^{\mathrm{des}} - \boldsymbol{p}_e \\ \boldsymbol{e}_{\omega_e} \end{bmatrix}, \tag{18}$$

and $W^{\#}$ is the pseudoinverse of W, i.e. $W^{\#} = W^T\left(WW^T\right)^{-1}$, and $*^{\mathrm{des}}$ ($* = \boldsymbol{r}_0, \boldsymbol{p}_e, \boldsymbol{\omega}_0, \boldsymbol{\omega}_e$) is the desired value of $*$, K_V and K_P are diagonal matrices consisting of scalar gain constants. Moreover,

$$\boldsymbol{e}_{\omega_*} = \frac{1}{2}\left(\boldsymbol{i}_* \times \boldsymbol{i}_{*_d} + \boldsymbol{j}_* \times \boldsymbol{j}_{*_d} + \boldsymbol{k}_* \times \boldsymbol{k}_{*_d}\right)\ (* = 0, e) \tag{19}$$

where $\boldsymbol{i}_*$, $\boldsymbol{j}_*$ and $\boldsymbol{k}_*$ are unit vectors along the axes of Σ_* with respect to Σ_I, and these vectors can be obtained from the rotational matrix (Luh et al., 1980):

$$\begin{bmatrix} \boldsymbol{i}_* & \boldsymbol{j}_* & \boldsymbol{k}_* \end{bmatrix} = {}^I\boldsymbol{R}_* .$$

Using Equations (13) and (17) actual control input for UVMS is calculated by

$$\boldsymbol{u} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{\alpha}^{\mathrm{ref}} + \boldsymbol{N}(\boldsymbol{q},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{f} . \tag{20}$$

Here, we represent the matrices and vectors of Equation (13) to the block matrix form:

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}_{\mathrm{BB}} & \boldsymbol{M}_{\mathrm{BM}} \\ \boldsymbol{M}_{\mathrm{MB}} & \boldsymbol{M}_{\mathrm{MM}} \end{bmatrix}, \quad \boldsymbol{N} = \begin{bmatrix} \boldsymbol{N}_{\mathrm{BB}} & \boldsymbol{N}_{\mathrm{BM}} \\ \boldsymbol{N}_{\mathrm{MB}} & \boldsymbol{N}_{\mathrm{MM}} \end{bmatrix}, \quad \boldsymbol{f}_D = \begin{bmatrix} \boldsymbol{f}_{\mathrm{B}} \\ \boldsymbol{f}_{\mathrm{M}} \end{bmatrix}, \quad \boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_{\mathrm{B}} \\ \boldsymbol{u}_{\mathrm{M}} \end{bmatrix}.$$

Then we have the following equation with respect to the input of the vehicle:

$$\boldsymbol{M}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0 + \boldsymbol{M}_{\mathrm{BM}}\ddot{\boldsymbol{\phi}} + \boldsymbol{N}_{\mathrm{BB}}\boldsymbol{\nu}_0 + \boldsymbol{M}_{\mathrm{BM}}\dot{\boldsymbol{\phi}} + \boldsymbol{f}_{\mathrm{B}} = \boldsymbol{u}_{\mathrm{B}} . \tag{21}$$

And the time derivative of Equation (8) is

$$\dot{\boldsymbol{s}} = \boldsymbol{C}\dot{\boldsymbol{\nu}}_0 + \boldsymbol{D}\ddot{\boldsymbol{\phi}} + \dot{\boldsymbol{C}}\boldsymbol{\nu}_0 + \dot{\boldsymbol{D}}\dot{\boldsymbol{\phi}} . \tag{22}$$

Comparing with Equations (21) and (22), $\boldsymbol{C} = \boldsymbol{M}_{\mathrm{BB}}$, $\boldsymbol{D} = \boldsymbol{M}_{\mathrm{BM}}$, $\dot{\boldsymbol{C}} = \boldsymbol{N}_{\mathrm{BB}}$, $\dot{\boldsymbol{D}} = \boldsymbol{N}_{\mathrm{BM}}$ and $\dot{\boldsymbol{s}} = \boldsymbol{u}_{\mathrm{B}} - \boldsymbol{f}_{\mathrm{B}}$ are obtained. Moreover,

$$\dot{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{0} & -\left(\tilde{\dot{\boldsymbol{p}}}_e - \tilde{\dot{\boldsymbol{r}}}_0\right) \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix},$$

$$\dot{\boldsymbol{B}} = \begin{bmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \cdots & \boldsymbol{b}_n \\ \boldsymbol{\omega}_1 \times \boldsymbol{k}_1 & \boldsymbol{\omega}_1 \times \boldsymbol{k}_2 & \cdots & \boldsymbol{\omega}_1 \times \boldsymbol{k}_n \end{bmatrix}$$

where $\boldsymbol{b}_i = (\boldsymbol{\omega}_i \times \boldsymbol{k}_i) \times (\boldsymbol{p}_e - \boldsymbol{p}_i) + \tilde{\boldsymbol{k}}_i(\dot{\boldsymbol{p}}_e - \dot{\boldsymbol{p}}_i)$. Therefore, all elements of $\boldsymbol{W}$ and $\dot{\boldsymbol{W}}$ in Equation (16) can be calculated.

3.2 Disturbance compensation of vehicle

From the viewpoint of underwater robot control, parameters and coefficients of hydrodynamic models are generally used as constant values that depend on the shape of robots (Fossen, 1995). The RAC law (17) can reduce the influence of the modelling errors of hydrodynamics by position and velocity feedbacks. Here, to obtain higher control performance, the influence of hydrodynamic modelling error with respect to the vehicle is treated as a disturbance and a disturbance compensation method is introduced.

First, the basic disturbance compensation is described. For $\boldsymbol{M}_{\mathrm{BB}}$ in Equation (21) the nominal model using constant values of added mass, added moment of inertia and drag coefficient is defined as $\overline{\boldsymbol{M}}_{\mathrm{BB}}$. Moreover, the basic disturbance is defined as

$$\boldsymbol{f}_L = \boldsymbol{u}_{\mathrm{B}} - \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0 , \tag{23}$$

and the estimated value is calculated by

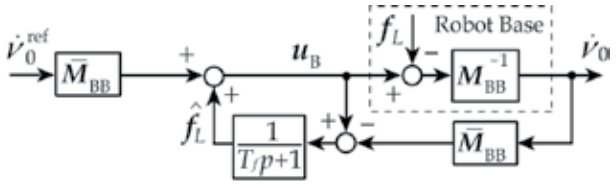
$$\hat{f}_L = F(p)\left(\boldsymbol{u}_{\mathrm{B}} - \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0\right) \tag{24}$$

where $F(p) = 1/(T_f p + 1)$ is a low pass filter with a time constant T_f and p is the time differentiation operator.

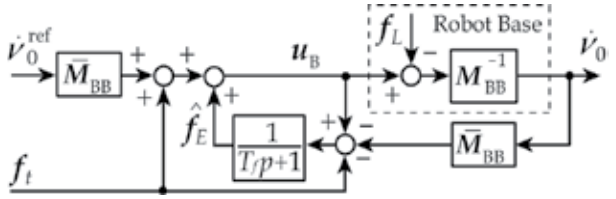
Therefore, for the reference acceleration of vehicle $\dot{\boldsymbol{\nu}}_0^{\mathrm{ref}}$, the control input with disturbance compensation becomes

$$\boldsymbol{u}_{\mathrm{B}} = \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0^{\mathrm{ref}} + \hat{f}_L, \tag{25}$$

and the configuration of the basic disturbance compensation is shown in Figure 3(a).



(a) Basic disturbance compensation



(b) Modified disturbance compensation

Fig. 3. Configuration of disturbance compensation

Next, the basic disturbance compensation is modified. For $\boldsymbol{M}_{\mathrm{BM}}$, $\boldsymbol{N}_{\mathrm{BB}}$, $\boldsymbol{N}_{\mathrm{BM}}$ and $\boldsymbol{f}_{\mathrm{B}}$ in Equation (21) the nominal models using constant values of added mass, added moment of inertia and drag coefficient are defined as $\overline{\boldsymbol{M}}_{\mathrm{BM}}$, $\overline{\boldsymbol{N}}_{\mathrm{BB}}$, $\overline{\boldsymbol{N}}_{\mathrm{BM}}$ and $\bar{\boldsymbol{f}}_{\mathrm{B}}$, respectively. Then the vehicle control input with these nominal models and the reference acceleration $\boldsymbol{\alpha}^{\mathrm{ref}}$ becomes

$$\overline{\boldsymbol{u}}_{\mathrm{B}} = \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0^{\mathrm{ref}} + f_t \tag{26}$$

where

$$f_t = \overline{\boldsymbol{M}}_{\mathrm{BM}}\ddot{\boldsymbol{\phi}}^{\mathrm{ref}} + \overline{\boldsymbol{N}}_{\mathrm{BB}}\boldsymbol{\nu}_0 + \overline{\boldsymbol{N}}_{\mathrm{BM}}\dot{\boldsymbol{\phi}} + \bar{\boldsymbol{f}}_{\mathrm{B}}. \tag{27}$$

From Equations (23) and (26) the modelling error with respect to the hydrodynamics can be defined as

$$f_E = \boldsymbol{u}_{\mathrm{B}} - \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0 - f_t, \tag{28}$$

and the estimated value is calculated by

$$\hat{f}_E = F(p)\left(\boldsymbol{u}_{\mathrm{B}} - \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{\nu}}_0 - f_t\right). \tag{29}$$

Therefore, the control input with disturbance compensation becomes

$$\boldsymbol{u}_{\mathrm{B}} = \overline{\boldsymbol{M}}_{\mathrm{BB}}\dot{\boldsymbol{v}}_0^{\mathrm{ref}} + \boldsymbol{f}_t + \hat{\boldsymbol{f}}_L \tag{30}$$

and the configuration of the modified disturbance compensation is shown in Figure 3(b).

4. Experiment of continuous-time RAC

In this section, some experiments of the RAC method are done for the vertical type 2-link underwater robot shown in Figure 1.

4.1 Experimental system

Figure 4 shows the configuration of the experimental system. A robot has a 2-DOF manipulator with joints that are actively rotated by velocity and torque control type servo actuators consisting of servo motors and incremental type encoders. The physical parameters of the underwater robot are shown in Table 1. Moreover, four 40[W] thrusters are attached to the vertical and horizontal directions on the robot base to provide propulsion for controlling the position and attitude angle of the base. The forward and reverse propulsion generated by the thruster are calculated by

$$F = \begin{cases} 1.341v^2 - 1.363v - 0.026 & (1.2 \le v \le 4) \\ -0.763v^2 - 0.835v + 0.019 & (-4 \le v \le -1.2) \end{cases} \tag{31}$$

where v is the input voltage to the power amplifier of the thruster. Note that Equation (31) were obtained from the experiments (Yamada & Sagara, 2002).

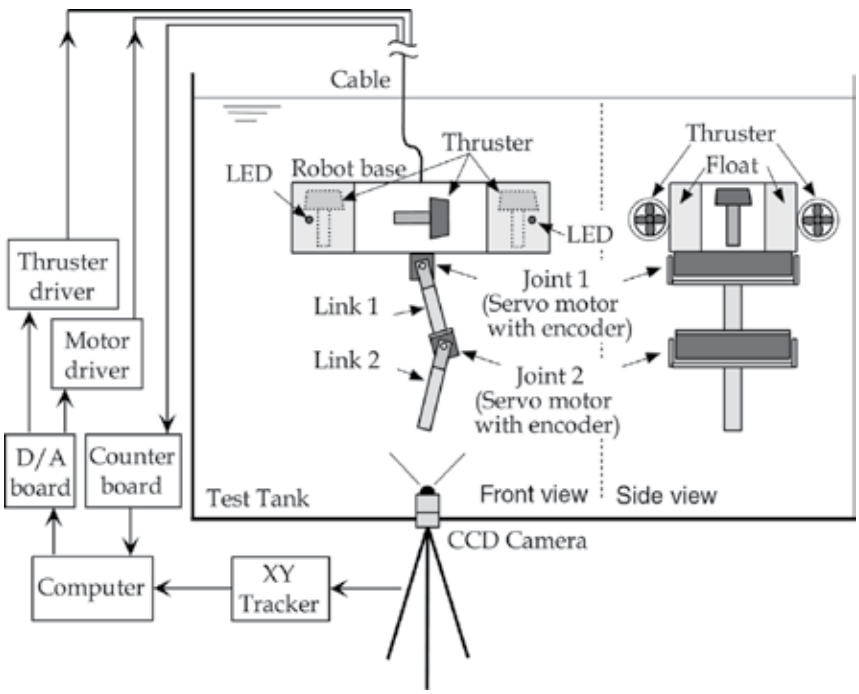


Fig. 4. Configuration of the underwater robot system

The measurement and control system consist of a CCD camera, a video tracker, and a personal computer (PC). Two LEDs are attached to the base, and their motion is monitored by the CCD camera. Video signals of the LED markers are transformed into position data by the video tracker, and put into the PC via a GPIB communication line. Using the position data and the rotational angle of each joint measured by the encoder, the positions and attitude angles of the robot base and manipulator are computed in the PC. The PC is also used as a controller.

	Base	Link 1	Link 2
Mass (kg)	26.04	4.25	1.23
Moment of inertia ($\mathrm{kg}\cdot\mathrm{m}^2$)	1.33	0.19	0.012
Link length (x_i direction) (m)	0.2	0.25	0.25
Link length (z_i direction) (m)	0.81	0.04	0.04
Link width (m)	0.42	0.12	0.12
Added mass (x_i direction) (kg)	72.7	1.31	0.1
Added mass (z_i direction) (kg)	6.28	3.57	2.83
Added moment of inertia ($\mathrm{kg}\cdot\mathrm{m}^2$)	1.05	0.11	0.06
Drag coefficient (x_i direction)	1.2	0	0
Drag coefficient (z_i direction)	1.2	1.2	1.2

Table 1. Physical parameters of underwater robot

4.1 Comparison of control performance of RAC and computed torque methods

In this subsection, to compare the control performances of the RAC method and a computed torque method that is generally used to control of UVMS, simulations and experiments are done. Note that joint torque control type servo actuators are used in the experiments.

Model of vertical type 2-link underwater robot is shown in Figure 5. In this figure F_i (i = 1, 2, 3) is the thrust of thruster and R is a distance from the origin of Σ_0 to the thruster. For the model shown in Figure 5 kinematic, momentum and dynamic Equations (3), (8) and (13) are reduced to

$$\dot{\boldsymbol{p}}_{e_V} = \boldsymbol{A}_V \dot{\boldsymbol{x}}_{0_V} + \boldsymbol{B}_V \dot{\boldsymbol{\phi}}_V , \tag{32}$$

$$\boldsymbol{s}_V = \boldsymbol{C}_V \dot{\boldsymbol{x}}_{0_V} + \boldsymbol{D}_V \dot{\boldsymbol{\phi}}_V , \tag{33}$$

$$\boldsymbol{M}_V(\boldsymbol{q}_V)\ddot{\boldsymbol{q}}_V + \boldsymbol{N}_V(\boldsymbol{q}_V, \dot{\boldsymbol{q}}_V)\dot{\boldsymbol{q}}_V + \boldsymbol{f}_V = \boldsymbol{u}_V \tag{34}$$

where

$$\boldsymbol{p}_{e_V} = \begin{bmatrix} p_{e_x} \\ p_{e_z} \end{bmatrix},\ \boldsymbol{x}_{0_V} = \begin{bmatrix} r_{0_x} \\ r_{0_z} \\ \phi_0 \end{bmatrix},\ \boldsymbol{\phi}_V = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix},\ \boldsymbol{q}_V = \begin{bmatrix} \boldsymbol{x}_{0_V} \\ \boldsymbol{\phi}_V \end{bmatrix},\ \boldsymbol{u}_V = \begin{bmatrix} \boldsymbol{u}_{\mathrm{B}_V} \\ \boldsymbol{u}_{\mathrm{M}_V} \end{bmatrix},\ \boldsymbol{u}_{\mathrm{B}_V} = \begin{bmatrix} f_{0_x} \\ f_{0_z} \\ \tau_0 \end{bmatrix},\ \boldsymbol{u}_{\mathrm{M}_V} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix},$$

and $\boldsymbol{A}_V$, $\boldsymbol{B}_V$, $\boldsymbol{D}_V$, $\boldsymbol{C}_V$ and $\boldsymbol{s}_V$ are appropriate matrices and vector.

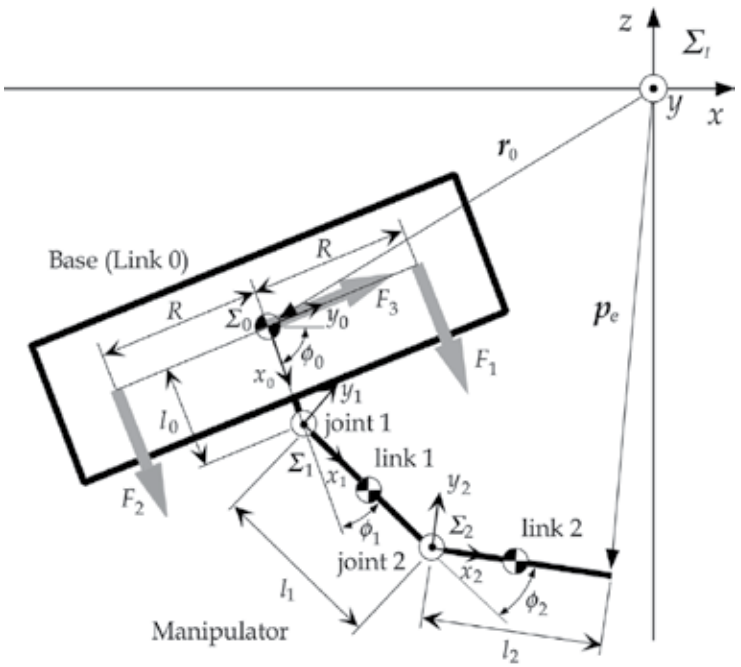


Fig. 5. Model of vertical type 2-link underwater robot

Similarly, the RAC law (17) is reduced to

$$\boldsymbol{\alpha}_V^{\mathrm{ref}}(t) = \boldsymbol{W}_V{}^{\#}(t)\{\boldsymbol{\beta}_V^{\mathrm{ref}}(t) + \boldsymbol{\gamma}_V(t) - \dot{\boldsymbol{W}}_V(t)\dot{\boldsymbol{q}}_V(t)\} \tag{35}$$

where

$$\boldsymbol{\beta}_V^{\mathrm{ref}} = \begin{bmatrix} \ddot{\boldsymbol{x}}_{0_V}^{\mathrm{des}} \\ \ddot{\boldsymbol{p}}_{e_V}^{\mathrm{des}} \end{bmatrix} + \boldsymbol{K}_{V_V} \begin{bmatrix} \dot{\boldsymbol{x}}_{0_V}^{\mathrm{des}} - \dot{\boldsymbol{x}}_{0_V} \\ \dot{\boldsymbol{p}}_{e_V}^{\mathrm{des}} - \dot{\boldsymbol{p}}_{e_V} \end{bmatrix} + \boldsymbol{K}_{P_V} \begin{bmatrix} \boldsymbol{x}_{0_V}^{\mathrm{des}} - \boldsymbol{x}_{0_V} \\ \boldsymbol{p}_{e_V}^{\mathrm{des}} - \boldsymbol{p}_{e_V} \end{bmatrix}, \tag{36}$$

$$\boldsymbol{W}_V = \begin{bmatrix} \boldsymbol{C}_V + \boldsymbol{E}_3 & \boldsymbol{D}_V \\ \boldsymbol{A}_V & \boldsymbol{B}_V \end{bmatrix}, \quad \boldsymbol{\gamma}_V = \begin{bmatrix} \dot{\boldsymbol{s}}_V \\ \boldsymbol{0} \end{bmatrix},$$

and $\boldsymbol{\alpha}_V^{\mathrm{ref}}$ is the reference of $\boldsymbol{\alpha}_V (= \ddot{\boldsymbol{q}}_V)$, $\boldsymbol{K}_{V_V}$ and $\boldsymbol{K}_{P_V}$ are positive diagonal matrices.

On the other hand, a computed torque method is briefly described as follows. From Equation (32) the task-space velocity $\boldsymbol{\nu}_V = [\dot{\boldsymbol{x}}_{0_V}^T \quad \dot{\boldsymbol{p}}_{e_V}^T]^T$ and joint–space velocity $\dot{\boldsymbol{q}}_V$ are related as

$$\boldsymbol{\nu}_V(t) = \boldsymbol{J}(t)\dot{\boldsymbol{q}}_V(t) \tag{37}$$

where

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{0} \\ \boldsymbol{A}_V & \boldsymbol{B}_V \end{bmatrix}.$$

From Equation (37) the following equation can be obtained:

$$\ddot{\boldsymbol{q}}_V(t) = \boldsymbol{J}^{\#}(t)\{\dot{\boldsymbol{\nu}}_V(t) - \dot{\boldsymbol{J}}(t)\dot{\boldsymbol{q}}_V(t)\}. \tag{38}$$

For Equation (38) reference joint-space acceleration is defined as

$$\ddot{\boldsymbol{q}}_V^{\mathrm{ref}}(t) = \boldsymbol{J}^{\#}(t)\{\dot{\boldsymbol{\nu}}_V^{\mathrm{ref}}(t) - \dot{\boldsymbol{J}}(t)\boldsymbol{J}^{\#}(t)\boldsymbol{\nu}_V^{\mathrm{ref}}(t)\}. \tag{39}$$

Based on Equations (34) and (39) actual control input is calculated by using the following equation:

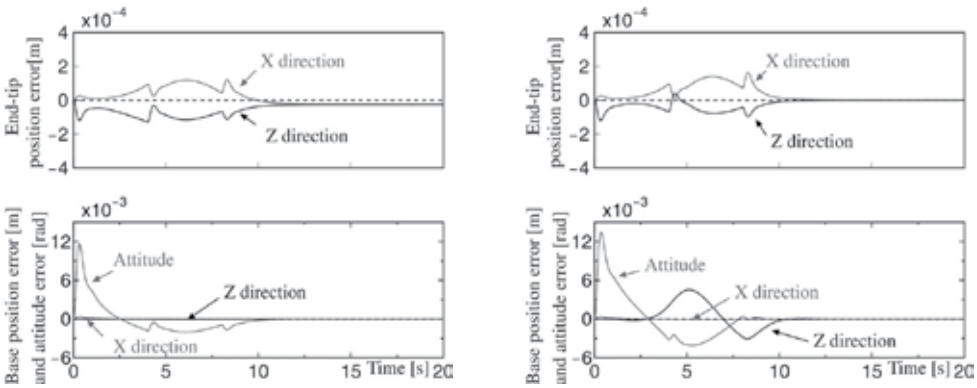
$$\boldsymbol{M}_V\ddot{\boldsymbol{q}}_V^{\mathrm{des}} + \boldsymbol{N}_V\dot{\boldsymbol{q}}_V + \boldsymbol{f}_V = \boldsymbol{u}_V \tag{40}$$

where

$$\boldsymbol{q}_V^{\mathrm{des}} = \boldsymbol{q}_V^{\mathrm{ref}} + \overline{\boldsymbol{K}}_{V_V}(\dot{\boldsymbol{q}}_V^{\mathrm{ref}} - \dot{\boldsymbol{q}}_V) + \overline{\boldsymbol{K}}_{P_V}(\boldsymbol{q}_V^{\mathrm{ref}} - \boldsymbol{q}_V), \tag{41}$$

and $\overline{\boldsymbol{K}}_{V_V}$ and $\overline{\boldsymbol{K}}_{P_V}$ are positive diagonal matrices.

Both simulations and experiments are carried out under the following condition. The desired end-tip position is set up along a straight path from the initial position to the target. On the other hand, the desired position and attitude of the base are set up the initial values. The feedback gains are $\boldsymbol{K}_{V_V} = \overline{\boldsymbol{K}}_{V_V} = \mathrm{diag}\{10 \quad 10 \quad 10 \quad 10 \quad 10\}$ and $\boldsymbol{K}_{P_V} = \overline{\boldsymbol{K}}_{P_V} = \mathrm{diag}\{100 \quad 100 \quad 100 \quad 50 \quad 50\}$. The initial relative joint angles are $\phi_0 = -\pi/2$ [rad], $\phi_1 = \pi/3$ [rad] and $\phi_2 = -5\pi/18$ [rad].



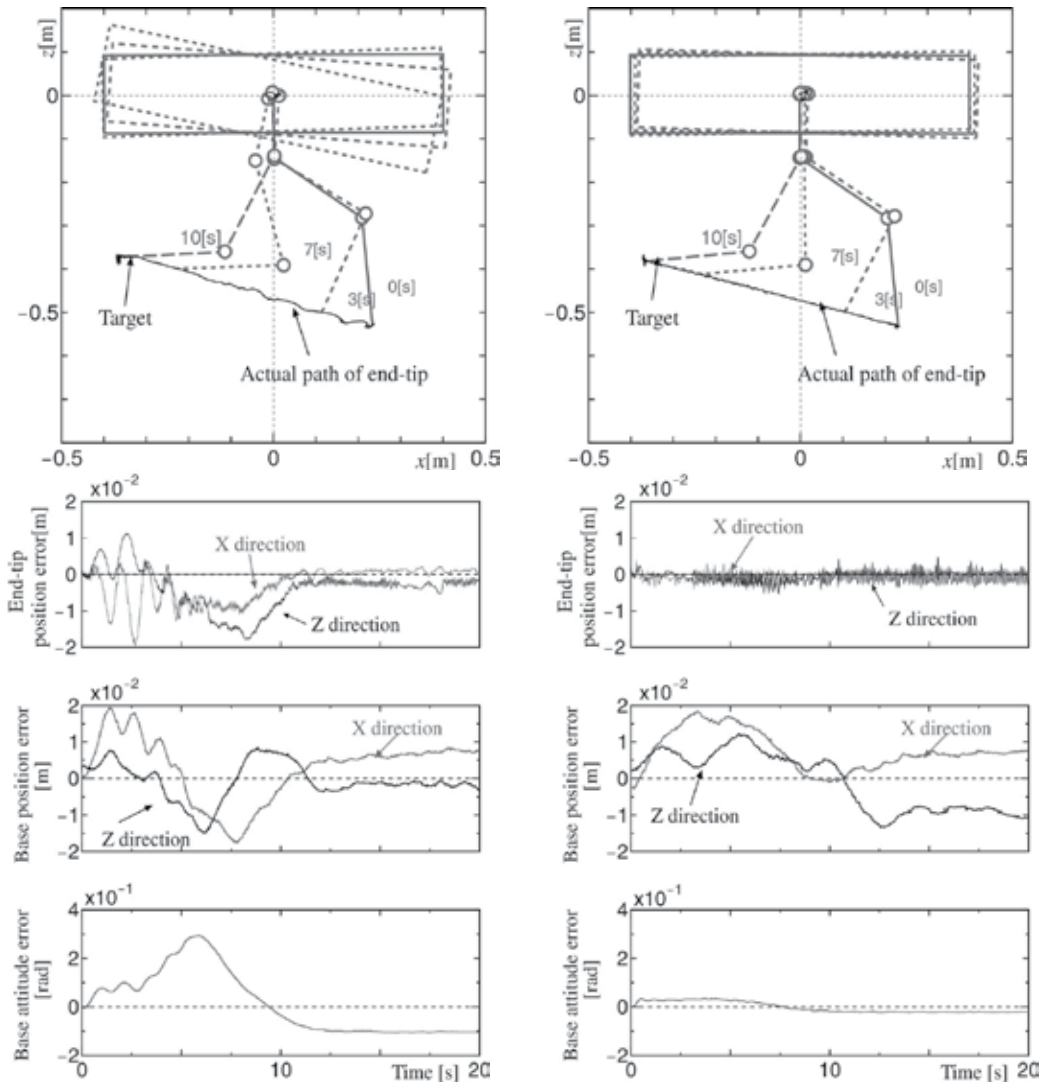
(a) Computed torque method (b) RAC method

Fig. 6. Simulation results of computed torque method and RAC method

First, simulation results of the computed torque method and the RAC method are shown in Figure 6(a) and (b). From Figure 6 we can see that both control methods have similar performance.

Next, we show the experimental results. As a computer is used for a controller in experiments, the sampling period for the controller is set up to $T = 1/60$ [s]. Figure 7 shows the both experimental results. From this figure, we can see that the performance of the computed torque method becomes worse. Since the computed torque method only uses joint-space errors, the control performance of the end-tip of the manipulator depends on the

robot base (vehicle) control performance. Therefore, if the acceleration and velocity relations between the end-tip and joints are inaccurate or the control performance of the vehicle is not better, good control performance of the end-tip cannot be obtained. On the other hand, from Figure 7 it can be seen that the RAC method has good control performance.



(a) Computed torque method (b) RAC method

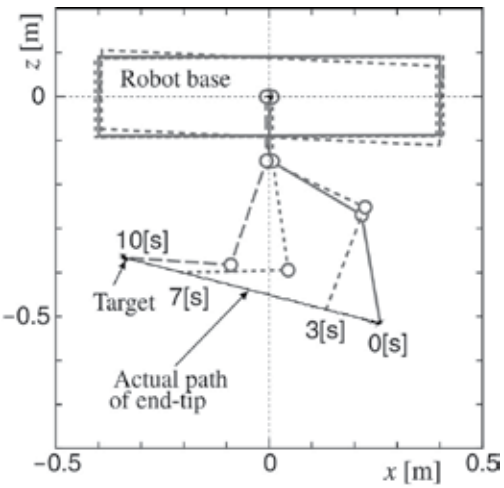
Fig. 7. Experimental results of computed torque method and RAC method

4.2 RAC method with disturbance compensation of vehicle

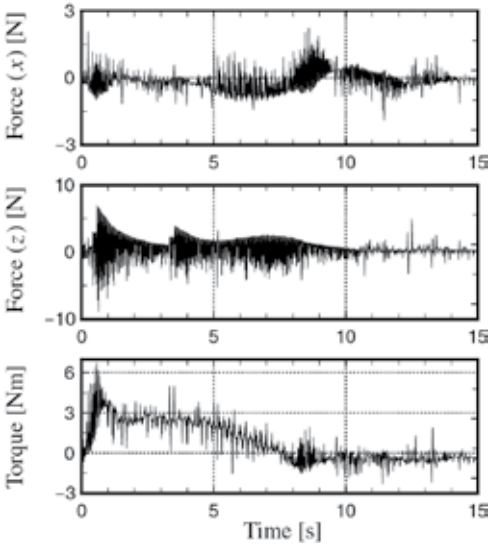
Experiments are carried out under the following condition. The desired end-tip position is set up along a straight path from the initial position to the target. On the other hand, the desired position and attitude of the base are set up the initial values. The feedback gains are

$\boldsymbol{K}_{V_V} = \mathrm{diag}\{10 \quad 10 \quad 10 \quad 20 \quad 20\}$ and $\boldsymbol{K}_{P_V} = \mathrm{diag}\{100 \quad 100 \quad 100 \quad 100 \quad 100\}$. The time constant of filter is $T_f = 1$ [s]. In this case joint velocity control type actuators are used.

Figure 8 shows the motion of the robot and estimated disturbance of the RAC with disturbance compensation, and Figure 9 shows the time histories of experimental results with and without disturbance compensation. Form Figures 8 and 9, it can be seen that the end-tip of manipulator follows the desired trajectory. Moreover, since the robot base position and attitude errors become small values using the disturbance compensation, the end-tip position error is also reduced.

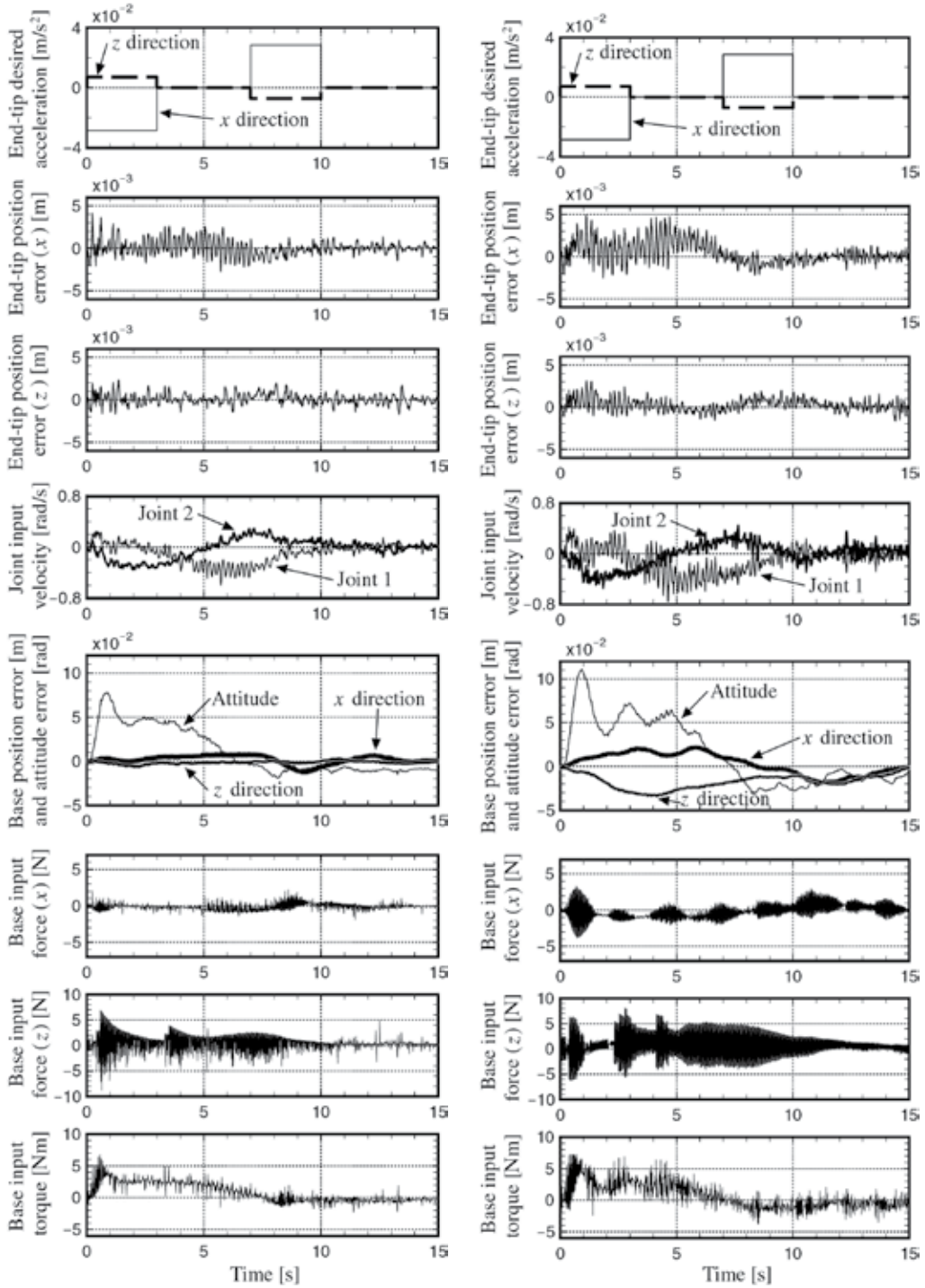


(a) Motion



(b) Estimated disturbance

Fig. 8. Experimental result of RAC method with disturbance compensation



(a) RAC with disturbance compensation (b) RAC without disturbance compensation

Fig. 9. Experimental results of RAC method with and without disturbance compensation

5. Discrete-time RAC

In practical systems digital computers are utilized for controllers, but there is no discrete-time control method for UVMS except our proposed methods (Sagara, 2003; Sagara et al., 2004; Sagara et al., 2006; Yatoh & Sagara, 2008). In this section, we address discrete time RAC methods including the ways of disturbance compensation of the vehicle and avoidance of singular configuration of the manipulator.

5.1 Discrete-time RAC law

Discretizing Equation (16) by a sampling period T, and applying $\boldsymbol{\beta}(k)$ and $\dot{\boldsymbol{W}}(k)$ to the backward Euler approximation, the following equation can be obtained:

$$\boldsymbol{W}(k)\boldsymbol{\alpha}(k-1)=\frac{1}{T}\{\boldsymbol{\nu}(k)-\boldsymbol{\nu}(k-1)+T\boldsymbol{\gamma}(k)-[\boldsymbol{W}(k)-\boldsymbol{W}(k-1)]\boldsymbol{\zeta}(t)\} \tag{42}$$

where $\boldsymbol{\nu}=[\boldsymbol{\nu}_0^T \quad \boldsymbol{\nu}_e^T]^T$. Note that a computational time delay is introduced into Equation (42), and the discrete time kT is abbreviated to k.
For Equation (42), the desired acceleration is defined as

$$\boldsymbol{\alpha}_d(k)=\frac{1}{T}\boldsymbol{W}^{\#}(k)\{\boldsymbol{\nu}_d(k+1)-\boldsymbol{\nu}_d(k)+\boldsymbol{\Lambda}\boldsymbol{e}_\nu(k)+T\boldsymbol{\gamma}(k)\} \tag{43}$$

where

$$\boldsymbol{e}_\nu(k)=\boldsymbol{\nu}_d(k)-\boldsymbol{\nu}(k) \tag{44}$$

and $\boldsymbol{\nu}_d(k)$ is the desired value of $\boldsymbol{\nu}(k)$, $\boldsymbol{\Lambda}=\mathrm{diag}\{\lambda_i\}$ ($i=1,\cdots,12$) is the velocity feedback gain matrix.
From Equations (42) and (43) we have

$$T\boldsymbol{W}(k)\boldsymbol{e}_\alpha(k-1)=\boldsymbol{e}_\nu(k)-\boldsymbol{e}_\nu(k-1)+\boldsymbol{\Lambda}\boldsymbol{e}_\nu(k)-T\{\boldsymbol{\gamma}(k)-\boldsymbol{\gamma}(k-1)\}+\{\boldsymbol{W}(k)-\boldsymbol{W}(k-1)\}\boldsymbol{\zeta}(t) \tag{45}$$

where

$$\boldsymbol{e}_\alpha(k)=\boldsymbol{\alpha}_d(k)-\boldsymbol{\alpha}(k).$$

Assuming $\boldsymbol{W}(k)\approx\boldsymbol{W}(k-1)$ and $\boldsymbol{\gamma}(k)\approx\boldsymbol{\gamma}(k-1)$ for one sampling period, Equation (45) can be rewritten as

$$T\boldsymbol{W}(k)\boldsymbol{e}_\alpha(k-1)=\{(q-1)\boldsymbol{E}_{12}+\boldsymbol{\Lambda}\}\boldsymbol{e}_\nu(k) \tag{46}$$

where q is the forward shift operator. Since all elements of $\boldsymbol{W}(k)$ are bounded, if λ_i is selected to satisfy $0<\lambda_i<1$ and the convergence of $\boldsymbol{e}_\alpha(k)$ tends to zero as k tends to infinity, the convergence of $\boldsymbol{e}_\nu(k)$ to zero as k tends to infinity can be ensured from Equation (46).
Moreover, the desired velocity of $\boldsymbol{\nu}(k)$ is defined as

$$\boldsymbol{\nu}_d(k)=\frac{1}{T}\boldsymbol{S}_{0e}(k)\{\boldsymbol{x}_d(k)-\boldsymbol{x}_d(k-1)+\boldsymbol{\Gamma}\boldsymbol{e}_x(k-1)\} \tag{47}$$

where

$$S_{0e} = \begin{bmatrix} \boldsymbol{E}_3 & & \boldsymbol{0} & \\ & S_{\psi_0} & & \\ & \boldsymbol{0} & \boldsymbol{E}_3 & \\ & & & S_{\psi_e} \end{bmatrix}, \quad \boldsymbol{e}_x(k) = \boldsymbol{x}_d(k) - \boldsymbol{x}(k), \quad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{r}_0 \\ \boldsymbol{\psi}_0 \\ \boldsymbol{p}_e \\ \boldsymbol{\psi}_e \end{bmatrix},$$

and $\boldsymbol{x}_d(k)$ is the desired value of $\boldsymbol{x}(k)$ ($=[\boldsymbol{x}_0^T \quad \boldsymbol{x}_e^T]^T$), $\boldsymbol{\Gamma} = \mathrm{diag}\{\gamma_i\}$ ($i = 1, \cdots, 12$) is the position feedback gain matrix.
From Equations (44) - (47) the following equation can be obtained:

$$T\boldsymbol{e}_v(k) = S_{0e}\{\boldsymbol{E}_{12} - (\boldsymbol{E}_{12} - \boldsymbol{\Gamma})q^{-1}\}\boldsymbol{e}_x(k) \tag{48}$$

where $\boldsymbol{v}(k)$ is applied to the backward Euler approximation. From Equation (48), if γ_i is selected to satisfy $0 < \gamma_i < 1$ and the convergence of $\boldsymbol{e}_v(k)$ tends to zero as k tends to infinity, the convergence of $\boldsymbol{e}_x(k)$ to zero as k tends to infinity can be ensured.
The configuration of the control system described in this subsection is shown in Figure 10.

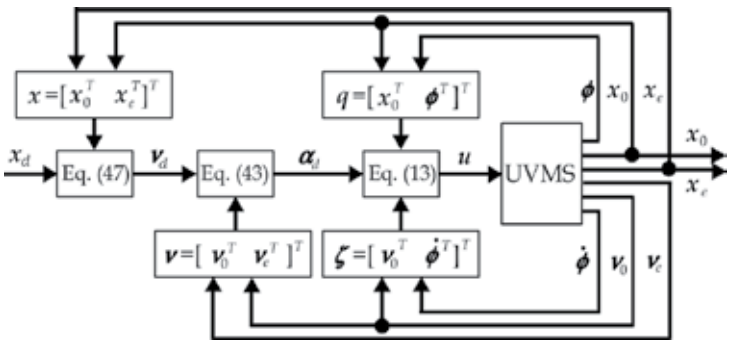


Fig. 10. Control system of discrete-time RAC

5.2 Disturbance compensation of vehicle

Discretizing the low pass filter, $F(p) = 1/(T_f p + 1)$, shown in Figure 3(b) (Godler et al., 2002), a digital version of disturbance compensation can be obtained. Figure 11 shows the digital version where $h = e^{-T_f/T}$.

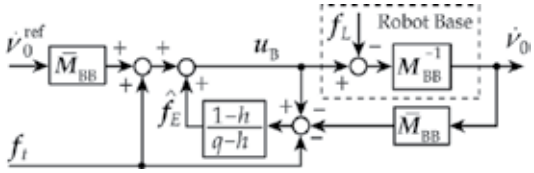


Fig. 11. Digital type disturbance compensation

5.3 Avoidance of a singular configuration

In much work on UVMS it is considered that the vehicle is keeping its initial state during the manipulation. In order to avoid the singular configuration of the manipulator in such case, the desired value of the vehicle is modified by using the determinant of the manipulator's Jacobian matrix $J(k) = \det \boldsymbol{J}(k)$ (Sagara et al., 2006).
The desired linear acceleration of the vehicle is defined as

$$\ddot{\boldsymbol{r}}_{0_d}(k) = \begin{cases} \dot{\boldsymbol{p}}_{e_d}(k) & (k_a \le k \le k_a + n_a) \\ \boldsymbol{0} & (\text{otherwise}) \\ -\dot{\boldsymbol{p}}_{e_d}(k) & (k_s \le k \le k_s + n_a) \end{cases} \tag{49}$$

where $\dot{\boldsymbol{p}}_{e_d}$ is the desired linear velocity of the end-tip of the manipulator, and k_aT and k_sT are the time when $|J(k)|$ becomes less or greater than a threshold J_s, respectively, and n_aT is the acceleration time.

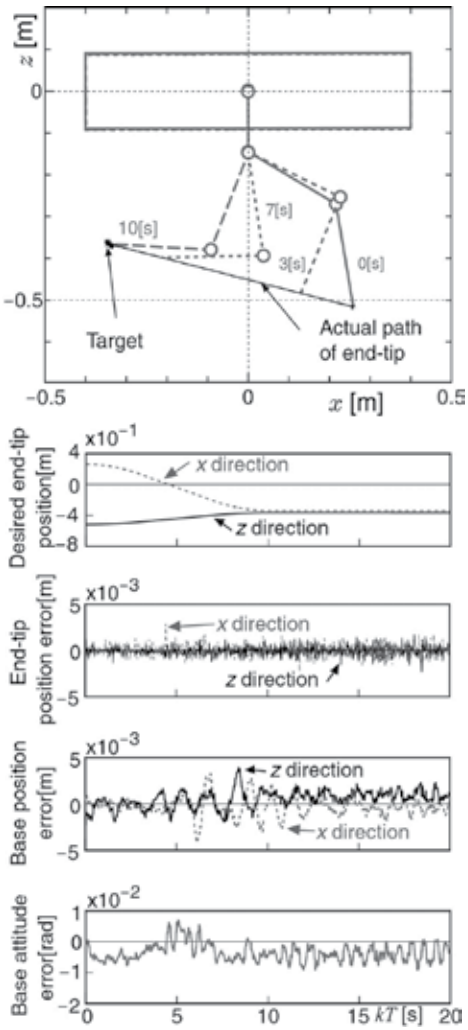


Fig. 12. Experimental result of discrete-time RAC

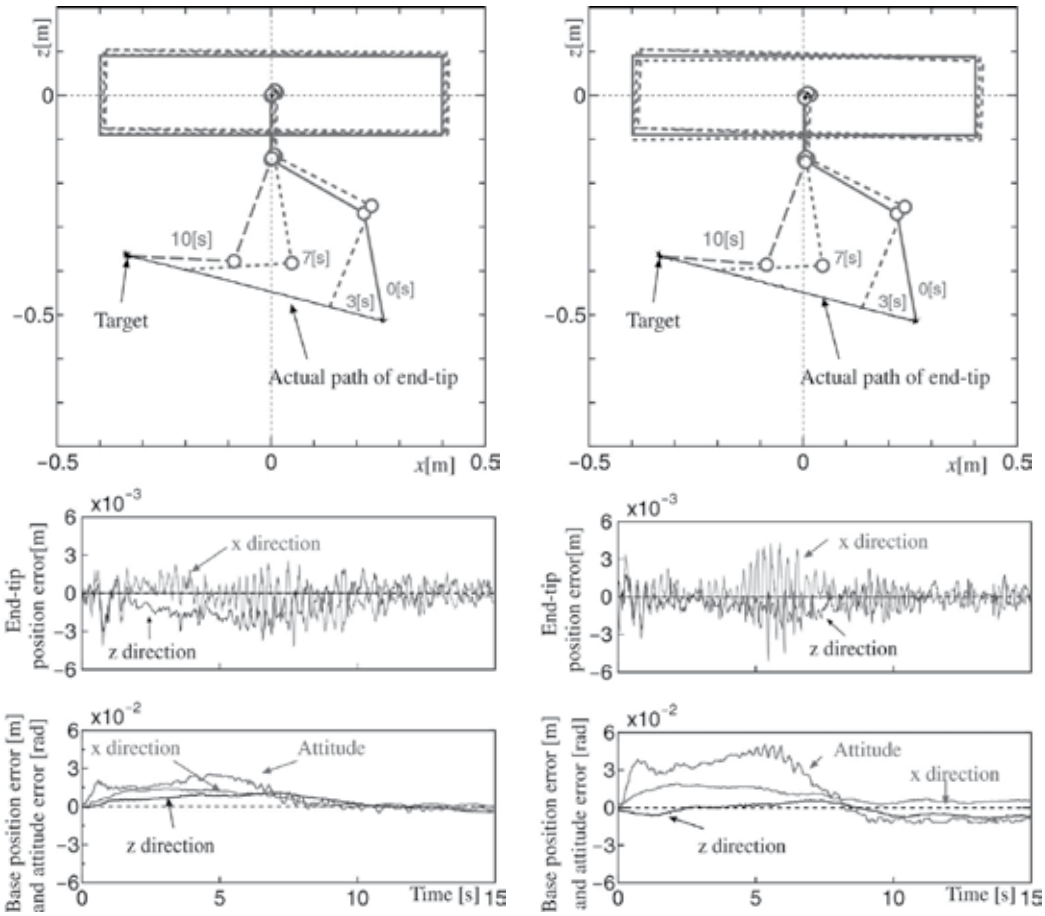
5.4 Experiment of discrete-time RAC

In this subsection, some experiments of the discrete-time RAC method described above are done for the underwater robot shown in Figures 1 and 4.

All experiments are carried out under the following condition. The desired end-tip position is set up along a straight path from the initial position to the target. On the other hand, the

desired position and attitude of the base are set up the initial values. The sampling period is $T = 1/60$ [s] based on the processing time of video tracker.

First, a basic discrete-time RAC experiment is done. In this case, the feedback gains are $\boldsymbol{\Lambda} = \mathrm{diag}\{0.6 \quad 0.6 \quad 0.25 \quad 0.25 \quad 0.25\}$ and $\boldsymbol{\Gamma} = \mathrm{diag}\{0.3 \quad 0.3 \quad 0.25 \quad 0.25 \quad 0.25\}$. Figure 12 shows the experimental result. From this figure, it can be seen that the discrete-time RAC method has good control performance and the performance is similar to that of the continuous-time version shown in Figure 7(b).



(a) RAC with disturbance compensation (b) RAC without disturbance compensation

Fig. 13. Experimental results of discrete-time RAC with and without disturbance compensation

Next, experiments of discrete-time RAC with and without disturbance compensation of the base are done. To validate the performance of disturbance compensation, the feedback gains of the RAC are $\boldsymbol{\Lambda} = \boldsymbol{\Gamma} = \mathrm{diag}\{0.3 \quad 0.3 \quad 0.2 \quad 0.2 \quad 0.2\}$. Using these values of the gains the basic control performance of the RAC becomes worse. The time constant of the filter for the disturbance compensation is $T_f = 0.1$ [s]. The experimental results of the RAC with and without disturbance compensation are shown in Figure 13(a) and (b), respectively. And Figure 14 shows the time history of the estimated disturbance. From Figures 13 and 14, it

can be seen that the position and attitude errors of the base are reduced by using the disturbance compensation.

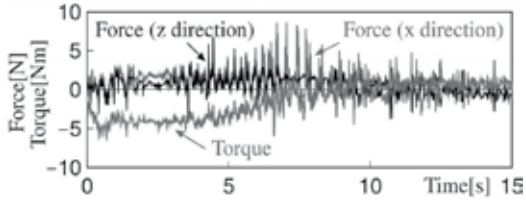


Fig. 14. Estimated disturbance (digital version)

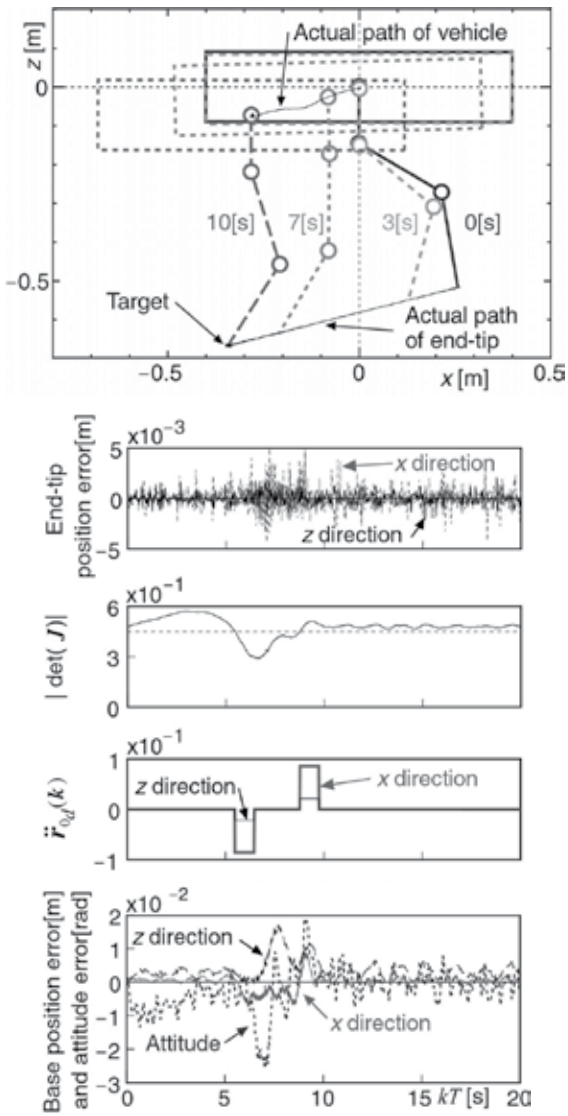


Fig. 15. Experimental result of discrete-time RAC considering singular configuration

Finally, an experiment of avoidance of singular configuration is done. In this case, the basic desired position and attitude of the base (vehicle) is set as the initial values, and the threshold of the determinant of the Jacobian matrix is $J_s = 0.45$. And the feedback gains are $\boldsymbol{\Lambda} = \boldsymbol{\Gamma} = \mathrm{diag}\{0.6 \quad 0.6 \quad 0.25 \quad 0.25 \quad 0.25\}$. The experimental result is shown in Figure 15. From Figure 15, we can see that the end-tip of the manipulator and base follow the desired trajectories avoiding the singular configuration of the manipulator and the tracking errors are very small.

6. Conclusion

In this chapter, our proposed continuous-time and discrete-time RAC methods was described and the both experimental results using a 2-link underwater robot were shown. For the continuous-time RAC method, experimental results showed that the RAC method has good control performance in comparison with a computed torque method and the RAC method with disturbance compensation can reduce the influence of the hydrodynamic modelling error. In practical systems digital computers are utilized for controllers. Then, we addressed discrete-time RAC methods including the ways of disturbance compensation and avoidance of singular configuration. Experimental results show that the control performance of the discrete-time RAC method is similar to the continuous version. Our future work is to carry out experiments in 3-dimensional space to evaluate the validity of the RAC methods.

7. References

Antonelli, G. & Chiaverini, S. (1998). Task-Priority Redundancy Resolution for Underwater Vehicle-Manipulator Systems, *Proceedings of the 1998 IEEE Iinternational Conference on Robotics and Automation*, pp. 768-773, 0-7803-4758-7, Leuven, May 1998

Antonelli, G.; Caccavale, F.; Chiaverini. S. & Villani, L. (2000). Tracking Control for Underwater Vehicle-Manipulator Systems with Velocity Estimation, *IEEE Journal of Oceanic Engineering*, Vol. 25, No. 3, pp. 399-413, 0364-9059

Antonelli, G. (2003). *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*, Springer-Verlag, 3-540-00054-2, Berlin

Fossen, T.I. (1994). *Guidance and Control of Ocean Vehicles*, John Wiley & Sons, 0-471-94113-1, NY

Godler, I.; Honda, H. & Ohnishi, K. (2002). Design Guidance for Disturbance Observer's Filter in Discrete Time, *Proceedings of 7th International Workshop on Advanced Motion Control*, pp. 390-395, 0-7803-7479-7, Maribor, Slovenia, Jul. 2002

Levesque, B. & Richard, M.J. (1994). Dynamic Analysis of a Manipulator in a Fluid Environment, *International Journal of Robotics Research*, Vol. 13, No. 3, pp. 221-231, 0278-3649

Luh, J.Y.S; Walker, M.W. & Paul, R.P.C. (1980). Resolved-Acceleration Control of Mechanical Manipulators, *IEEE Transactions on Automatic Control*, Vol. 25, No. 3, pp. 468-474, 0018-9286

Maheshi, H.; Yuh, J. & Lakshmi, R. (1991). A Coordinated Control of an Underwater Vehicle and Robotic Manipulator, J*ournal of Robotic Systems*, Vol. 8, No. 3, pp. 339-370, 07412223

McLain, T.W.; Rock, S.M. & Lee, M.J. (1996). Experiments in the Coordinated Control of an Underwater Arm/Vehicle System, In: *Underwater Robots*, Yuh, J.; Ura, T. & Bekey, G. A., (Ed), pp.137-158, Kluwer Academic Publishers, 0-7923-9754-1, MA

McLain, T.W. & Rock, S.M. (1998). Development and Experimental Validation of an Underwater Manipulator Hydrodynamic Model, *International Journal of Robotics Research*, Vol. 17, No. 7, pp. 748-759, 0278-3649

McMillan, S.; David, D.E. & McGhee, R.B. (1995). Efficient Dynamic Simulation of an Underwater Vehicle with a Robotic Manipulator, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 25, No, 8, pp. 1194-1206, 0018-9472

Sagara, S. (2003). Digital Control of an Underwater Robot with Vertical Planar 2-Link Manipulator, *Proceedings of the 8th International Symposium on Artificial Life and Robotics*, pp. 524-527, 4-9900462-3-4, Beppu, Jan. 2003

Sagara, S.; Shibuya. K. & Tamura, M. (2004). Experiment of Digital RAC for an Underwater Robot with Vertical Planar 2-Link Manipulator, *Proceedings of the 9th International Symposium on Artificial Life and Robotics*, pp. 337-340, 4-9900462-4-2, Beppu, Jan. 2004

Sagara, S.; Tamura, M.; Yatoh, T. & Shibuya, K. (2006). Digital RAC for Underwater Vehicle-Manipulator Systems Considering Singular Configuration, *Artificial Life and Robotics,* Vol. 10, No. 2, pp. 106-111, 1433-5298, Springer

Sarkar, N. & Podder, T.K. (2001). Coordinated Motion Planning and Control of Autonomous Underwater Vehicle-Manipulator Systems:Subject to Drag Optimization, *IEEE Journal of Oceanic Engineering*, Vol. 26, No. 2, pp. 228-239, 0364-9059

Tarn, T. J; Shoults, G.A. & Yang, S.P. (1996). A Dynamic Model of an Underwater Vehicle with a Robotic Manipulator Using Kane's Method, In: *Underwater Robots*, Yuh, J.; Ura, T. & Bekey, G. A., (Ed), pp.137-158, Kluwer Academic Publishers, 0-7923-9754-1, MA

Yamada, S. & Sagara, S. (2002). Resolved Acceleration Control of an Underwater Robot with Vertical Planar 2-Link Manipulator, *Proceedings of the 7th International Symposium on Artificial Life and Robotics*, pp. 230-233, 4-9900462-2-6, Beppu, Jan. 2002

Yatoh, T. & Sagara, S. (2007). Resolved Acceleration Control of Underwater Vehicle-Manipulator Systems Using Momentum Equation, *Proceedings of OCEANS 2007 MTS/IEEE Vancouver*, paper number 070427-004, 0-933957-35-1, Vancouver, Oct. 2007

Yatoh, T. & Sagara, S. (2008). Digital Type Disturbance Compensation Control of Underwater Vehicle-Manipulator Systems, *Proceedings of OCEANS'08 MTS/IEEE Kobe-Techno-Ocean'08*, paper number 071109-002, 978-1-4244-2126-8, Kobe, Apr. 2008

Yuh, J. (Ed). (1995). *Underwater Robotic Vehicles: Design and Control*, TSI Press, 0-9627451-6-2, NW

24

Studies on Hydrodynamic Propulsion of a Biomimetic Tuna

Afzal Suleman and Curran Crawford
University of Victoria
Canada

1. Introduction

Current unmanned undersea vehicles (UUVs) are almost exclusively propeller driven designs, which must inherently be optimized for a particular speed, sacrificing low speed manoeuvrability for cruising efficiency. Recently, biomimetic approaches to underwater vehicle propulsion have illuminated the exciting possibilities for performance improvements made possible by emulating fish motion. In particular, a number of test vehicles indicate that the carangiform swimming mode employed by highly developed species of fish, such as the Bluefin tuna, offers both a more efficient propulsion mechanism than propellers, in addition to the ability to perform quick manoeuvring. This book chapter presents studies on the propulsion efficiency of a biomimetic tuna at the University of Victoria. Two prototypes have been designed and implemented experimentally. The first prototype consists of a biomimetic tuna that employs shape memory alloy wires to affect shape induced propulsion. The second prototype propulsion model consists of four joints that are rotated using servomotors. Issues related to the mechanism, systems and energy are discussed. The performance and the lessons learned related to the two design philosophies are presented and discussed.

2. Background and motivation

Underwater vehicle design has in the past primarily focused on propeller driven designs. Research efforts have been directed towards optimizing propeller design for particular operating speeds, and also on improving manoeuvring performance through control surface, hull, and thruster configuration and design. Through computing power and mechanical system design improvements, unmanned undersea vehicles (UUVs) have been able to expand their operating envelope and carry out more ambitious, extended, and varied missions including oceanographic surveys, reconnaissance, cable laying, and mine hunting. Propeller driven designs inherently involve design tradeoffs; speed is traded for low speed manoeuvrability, and efficiency is balanced against operational speed range requirements.
Natural biological evolution has also struggled with the same design considerations, and produced extremely efficient modes of propulsion over millions of years of natural selection. The emerging field of biomimetics seeks to exploit this natural design process by copying the refined forms of living creatures found in nature. In the area of hydrodynamic propulsion, highly evolved species such as the Bluefin tuna employing the carangiform

swimming mode have been optimized for high speed cruising, while retaining excellent manoeuvring capabilities. Previous research, including the RoboTuna (Barret, 2000), RoboPike (Kumph, 2000), and VCUUV (Anderson & Kerrebrock, 2000), has proven the potential of emulating fish swimming in underwater vehicles, and suggest exciting possibilities for performance improvement over traditional UUV design in both efficiency and manoeuvrability. A study on the swimming modes for aquatic locomotion has been published by Stafiotakis et al (1999). Chiu et al. (2000, 2002) have analyzed and simulated the undulatory locomotion of a flexible slender body. Guo et al (2002) have proposed a method for coordinating body segments for controlling the motion of a biomimetic autonomous underwater vehicle. Barret et al (1996) used genetic algorithms to determine the optimal body motion of the RoboTuna. Harper et al (1998) and Blickhan and Chen (1994) have studied several methods for measuring the power of swimming fish. Due to the complexity of the fluid dynamics problem, analytic and computational analysis of the problem has not yet progressed to the point where proper simulation of the body and caudal fin is possible. Progress has been made though, starting with inviscid flow analysis in two dimensions (Lighthill, 1970), with current research efforts focusing on full CFD simulation of the entire fish with vorticity and turbulence.

Previous biomimetic fish designs have used conventional actuators, including cable drives, servomechanisms, and hydraulics. Development of multifunctional materials such as piezoelectrics, magnetostrictive materials, and shape memory alloys (SMAs) has presented an alternative actuation method. For this particular application, shape memory alloys are the most suitable of the multifunctional materials since they offer both the necessary strains and forces required for underwater propulsion.

Piezoelectrics may also find applications in such a vehicle for small shape changes in the fins or selective stiffening of the actuated structure. However, SMAs have a number of potential advantages, including simplicity, noiseless operation, and low driving voltages. They can be used as direct linear actuators, and do not require gearing systems, reducing system complexity and making them ideal for confined space applications. Since they are essentially solid-state electrical devices, they produce no acoustic signature, a valuable asset for some missions such as covert military operations and for sensitive acoustic measurements. The low driving voltages required are also suited to power supplies typically available on UUVs.

SMAs do have a number of drawbacks however, including low energy efficiency and performance degradation under repetitive operation beyond a couple of million cycles. The design of the SMA fish is such that new smart actuators could be easily integrated as they become available, in order to overcome some of these disadvantages. The "undulatory vehicle," has utilized SMAs for propulsion, however it was based on the hydrodynamically inefficient eel (Wardle & Reid, 1997). SMA actuators have also been successfully used for an actively controlled hydrofoil (Rediniotis, 1999).

The objective of this research is to develop a more efficient method of driving underwater vehicles using shape induced propulsion. The replacement of propellers with fish like locomotion is expected theoretically to offer 15-20% improvements in efficiency. Fish like propulsion should offer more stealthy designs of vehicles. The following strategies for shape induced propulsion have been considered: (i) in the first instance, prototype I using SMA wires to induce shape control of the tuna body was concetpualized, manufactured and tested. These SMA devices act like little muscles. When they are heated they contract and when cooled they expand. A fish was designed using SMA wires as muscles to produce the

swimming action. This research resulted in some insightful observations based on the quantitative and qualitative results. The fish mechanism created measureable thrust while requiring large amounts of power for its operation. This finding suggested that a new approach was needed. The power requirement alone would mean that the fish would require a large battery source; (ii) next, the prototype II ServoTuna design was introduced to overcome these problems. This prototype was controlled by a single chip microcontroller which executed swimming motions using its four servomotors by autonomous control. The "travelling wave" type motion that is peculiar to fish was programmed into the single chip computer and it executed the correct wave pattern. The results obtained using the second generation prototype were more promising.

3. The shape memory alloy induced propulsion: prototype I

3.1 Design considerations

The design constraints initially placed on the vehicle included a 1m total length with 50% actuated length, typical of the biological carangiform swimming mode and previous biomimetic projects. In addition, some form of actuated surface for pitch and roll control was included, and provision for mounting a mast for tow tank testing was included.

In order to preserve the biomimetic approach to the design of the SMA fish and in the absence of detailed results from computational hull form optimisation, the profile of a real Bluefin tuna was used for the side profile. The Bluefin tuna was selected for its highly evolved cruising speed efficiency, and its carangiform swimming mode that in nature involves actuating approximately 50% of the fish's length. Two physical features of the tuna are though to contribute to its efficiency in addition to the swimming mode: the "necking" of the profile at the caudal peduncle, and the high aspect ratio caudal fin. The top profile followed extremely closely the shape of modern low drag airfoils, and consequently a NACA 63-015A airfoil section was chosen for this profile. The cross section of the entire hull is elliptical, again emulating nature. This cross section was chosen to reduce the bending stiffness of the hull in the posterior actuated segment, and also to provide the optimum hydrodynamic shape for the actuated segment while still retaining internal volume in the pressure hull. Patching a circular cross section forward to an aft elliptical cross section may be an option for future deep-diving designs, but would almost certainly compromise efficiency.

Examination of the biological tuna reveals that all of the fins except for the caudal fin fold into the body at cruising speed. For this reason, all of the fins except for the caudal and pectoral fins were excluded. The pectoral fins are essentially canards as they are located forward of the centre of mass, and will be used for both roll and pitch control, with modulation of the tail motion to be used for yaw control. For propulsion efficiency studies, these were not considered. Figure 1 illustrates the general layout of the SMA fish Prototype I.

Since the forward 50% of the body is non-actuated, it was constructed as a rigid shell and contains controllers, sensors, and a power source. For design simplicity, the nose cone is designed as a pressure hull in order to easily include on-board power and control later in the development cycle. The nose cone was constructed as a moulded fibreglass shell, with an aluminium bulkhead at the aft end, which mates to another bulkhead to which the actuated segment is attached. The two bulkheads were sealed together by means of an O-ring, allowing for easy access to the interior of the nose cone. The nose cone was reinforced

by aluminium ribs that would also include a hard point for mounting to the towing tank mast. In addition, the sealed pivots for the canards were located in the nose cone and attached to the ribs, along with the servos to control them. Any control electronics, internal ballast, and eventually batteries would also be attached to the ribs. This design has been successfully used in all of the other fish projects reported in the literature.

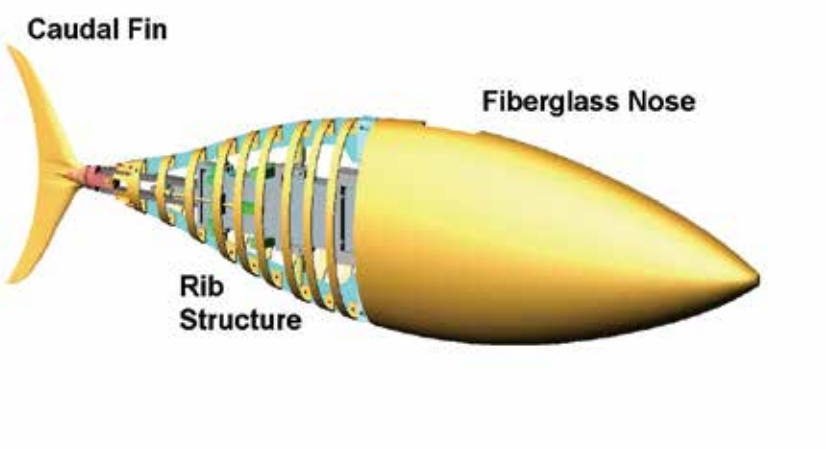


Fig. 1. The shape memory alloy based design

3.2 The actuated tail section

This section of the hull occupies the aft portion of the hull, and accounts for the other 50% of the hull, including the caudal fin. The caudal fin contributes the bulk of the propulsive force in the carangiform swimming mode that is being emulated, however the travelling wave that is present in the tail is also an important factor to the overall efficiency of the design, and therefore the tail was designed to be flexible. In order to achieve this, a "skeletal" structure was developed over which "skin" is stretched. The basic element of the skeleton is a spline element running from the bulkhead aft to the tail peduncle. This spline provides a smooth curving shape to the tail along its length. Ribs (again with elliptical cross section) were then attached to the spline to provide the 3-D form of tail. The entire tail section was flooded in order to avoid the complicated problem of sealing it, especially for deep diving missions. Flooding of the tail will also drastically increase the cycle frequency possible with the SMAs due to the increased heat convection off of the wires during cooling in water however at a higher energy cost.

A number of rib designs have been used in the past. Most are a variation on complete cross sections built out of aluminium (RoboTuna), plastic (undulatory vehicle), or foam (Draper Tuna), bolted to the spline. An interesting approach to the rib design problem was used on the RoboPike project; the ribs were formed out of fibreglass as a continuous helical spring, bolted along its length to the spline. In order to simplify the rib production and assembly process, the SMA fish uses "half-ribs" formed out of fibreglass. Since the ribs are formed in two halves, a single half mold was used to form the ribs for both sides of the fish. Mounting of the ribs on the spline was also simplified, since the ribs were simply bolted together through the spline without the need for bonding mounting blocks to the ribs or other complicated schemes. The ribs were placed approximately every 2.54 cm, and have a width of approximately 1 cm.

The skin material for the tail must allow the tail to flex while at the same time avoid deflections in the unsupported regions between ribs. The skin must also be impermeable so that fluid is forced to move along with the surface of the tail and thereby provide some of the thrust. Previous designs have all used Lycra as the skin material over a layer of reticulated foam or steel mesh avoid the problem of bulging between the ribs. The Lycra is by its nature impervious to water (there are also some surface treatments of the fabric available to reduce the permeability further) and has a very low elastic modulus. The layer of foam essentially forms a composite plate structure by increasing the bending stiffness of the fabric while not adversely increasing the tensile modulus that must be low to allow bending of the overall tail. Neoprene (wetsuit material) combines these functions, since it is actually comprised of a core of foam covered on both sides by Lycra fabric. It is available in thicknesses down to 1.5 mm, the same thickness used on modern high performance full body swimsuits that allow for a free range of motion of the swimmer. By aligning the fibres of the Lycra at 45° to the lateral axis of the fish the tensile modulus of the skin is minimized while retaining the bending stiffness of the fabric. The skin was hemmed at the forward end and the elasticity of the material holds it in a groove in the nose cone. At the tail end, the material wraps around the end of the caudal fin with a Velcro strip to hold it in place. There are two possible approaches to actuating the tail section using SMAs. In the robotic based design, it seeks to adapt SMA actuators to the designs that have been used in past projects that construct the tail as a robotic linkage. In these designs, the actuators effect rotation of the links, and the motion of the tail links is transmitted in some manner to the spline. In the RoboTuna, the tail has 6 actively controlled degrees of freedom, while in the VCUUV there are four. In these designs, as the links rotate with respect to one and other, the spline must extend. This is accomplished in the RoboTuna by means of a segmented spline, while in the case of the VCUUV follower rods mechanisms are used to transfer the motion of the links to the spline. Adapting this approach to the SMA fish, the concept shown in Figure 2 was developed with four wires per side of the joint. There are four controlled joints in this design, all with SMAs mounted in the "lever" configuration. This design allows the actuating wire to strain the wire on the opposing side, in order to set-up for the next cycle. In this way, the one-way shape memory effect can be used with no need for an external force to re-strain the wires after each cycle. The SMAs are secured to the links by Plexiglas clamp blocks at one end, and individually clamped with a bolt and washers at the other end to allow for adjust of tension in the each wire, as illustrated in Figure 2.

Initially, the links themselves were made of aluminium, with two ball bearings at each pivot point. The axles for the ball bearings also served as the shafts about which the spline blocks pivot, and were secured in the links with setscrews. Two e-clips along each pivot shaft retained the bearing and locate each spline block assembly. These blocks allowed the spline to extend during operation, and must themselves pivot about the same axes as the links to ensure smooth curvature of the spline. They were made of Delrin with stainless steel for the slider rods. One end of each of the slider rods has an e-clip positioned so that the extension of the spline is limited to avoid excessive local curvature of the spline segment. The opposite end of the rod is screwed into the other block. The spline is simply bolted to the spline blocks, and located in the groove in the blocks. The tail linkage and root spline segments are all bolted to the aft bulkhead. It was found that this approach was suited to conventional actuation methods such as cables and hydraulics, but somewhat defeats the purpose of utilizing SMAs to reduce the complexity of the vehicle.

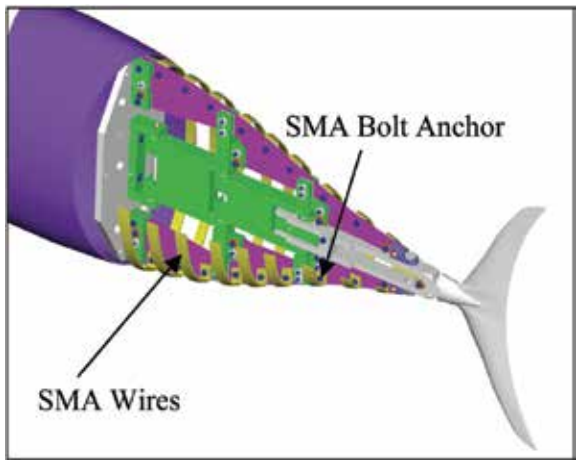


Fig. 2. The actuated tail section design

In an effort to adopt a more adaptive structures approach to the SMA fish, a new tail section was designed to take advantage of the SMA characteristics. In this design, the spline itself is actuated, much like the undulatory vehicle. The skeletal links are retained in this design, but their function changes to preventing torsion of the caudal fin about the lateral axis of the fish. In the absence of tail linkages, differential shedding of vortices off of the two tips of the fin would lead to a torsional moment on the spline, causing a change in the angle of attack of fin. This would lead to uncontrollable coupled pitching and yawing. It may be possible in future design to eliminate the need for tail linkages by using the torsional strength of the skin (note that the 45° orientation of the Lycra fibres increases the effective torsional stiffness of the tail) in conjunction with actuators at 45° to the lateral axis on the spline itself to control the torsion of the spline.

For the present prototype, the inclusion of the tail linkages greatly simplifies initial testing by eliminating torsional deformation from unbalanced forces on the tail, while not contributing undue complexity to the design. The links must contract in order for the spline to freely flex. This is accomplished in much the same manner as the spline sliders in the robotic design, using slider rods with the extent of travel limited by e-clips. The larger block is made of Delrin through which the rods will slide, while the other spine blocks are of aluminum. The rods thread into these blocks, and bearings are also located in the blocks, two per pivot axis. The aft spine linkage assembly is somewhat more complicated, since at the forward end it must be able to contract, and at the aft end be rigid for the set of SMAs controlling the caudal peduncle pivot. Note that this design again allows for three regions of the spline to be controlled, as well as the peduncle for a total of 4 DOF. The caudal fin was made in same manner as the canards with epoxy covered wood, laminated on a Plexiglas block at the peduncle pivot.

The spline is again attached to the links at the same pivot points as the links themselves by means of Delrin blocks that themselves pivot about the same axes as the links. The same blocks provide mounting points for the SMAs, clamped at the aft end between aluminium spacers and at the forward end by bolts and washers. The aluminium spacers are contained within grooves in the Delrin blocks and clamped with two bolts visible at the end of each spline block. The mounting of the SMAs for the caudal peduncle section is the same as outlined earlier. The cutouts in the spline serve to reduce its bending stiffness while

preserving the structural integrity for supporting the ribs. The spline was made of 0.8 mm Delrin sheet. The completed tail section prototype is shown in Figure 3.

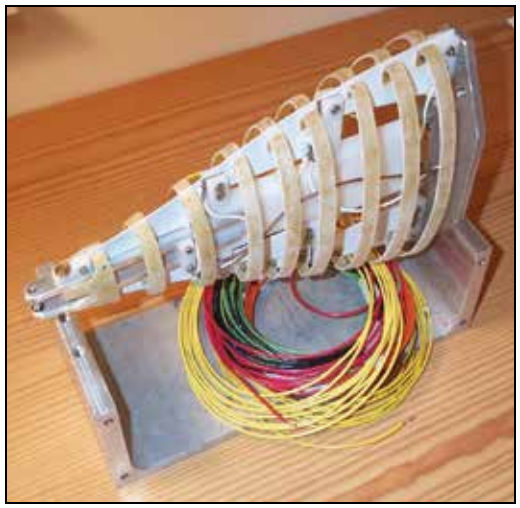


Fig. 3. The tail section prototype

4. SMA actuators

The shape memory effect is exhibited by a number of alloys, however the most common one is an alloy of Ni-Ti, referred to as Nitinol. The one-way shape memory effect occurs after an external force strains the material in its cold state. Upon heating, the material will return to the initial "remembered" shape, and if the material is constrained can produce a considerable force on the constraints. This effect is explained from a crystal structure approach by the following; the low temperature phase is martensite and possesses a low yield strength, which is easily plastically deformed by and external force. Upon heating past the transition temperature of the alloy, the phase of the material changes to austentite, a phase with high elastic modulus and yield strength, and due to the training process will attempt to return to the remembered shape. The transition temperature can be tailored by the alloying and heat treatment process to obtain a value between –100°C and 100°C. Nitinol can be obtained in a variety of forms, including thin film, rod and bar, tube, and wire stock. It is usually sold in its as formed state, and must undergo a complex heat treatment process in order to "train" it and thereby attain its memory. The process consists of heating and coiling cycles over which the material is strained from its remembered state, and must be repeated over 50 times.

Nitinol has been made available by Mondo-Tronics in pre-trained wire form, making integration into various robotics projects quite strait forward. The commercial name is either "Muscle Wires" or "Flexinol." The wires are available in a number of diameters ranging from 37μm up to 0.375mm, and are actuated by passing a current directly through the length of the wire. Larger wires can exert higher forces, however require correspondingly higher currents for Joule heating and are not capable of fast cycling. Low and high temperature wires are available with transition temperatures of 70°C and 90°C respectively, and care must be taken to avoid overheating which destroys the wires. The low temperature

wires were considered here due to the fact that they will be immersed in water that ensure adequate heat transfer for cooling. (Note that operating the wires in water is advertised to permit over 10 times the frequency of operation compared to operation in air due to increased convective cooling.) A number of the salient properties are listed in Table 1 below for the candidate wires. The Flexinol 300 series were selected to ensure adequate force is available for actuation.

FLEXINOL WIRE NAME	**150**	**250**	**300**
Wire diameter (Ωm)	150	250	300
Recommended current in air (mA)	400	1000	1750
Recommended recovery force (N)	3.2	9.1	12.3
Recommended deformation force (N)	0.6	1.7	2.4
Projected cycle rate in water (Hz)	3.3	1.5	1.2
Price/5m roll (US $)	75	79	88
Work output (J/g)	1		
Recommended deformation (%)	3-5		
Energy conversion efficiency (%)	5		
		Martensite	Austenite
Resistivity (μΩ cm)		76	82

Table 1. Flexinol Wire Properties

In the current model, four wires were used per side for each actuated segment of the tail. Using the wires in parallel linearly increases the force available while retaining the ability for fast cycle times, as opposed to using large diameter wires that would require much longer times for convective cooling of the wires. The wires and any contacting electrically conducting material was covered with PlastiDip for electrical insulation from the water.

4.1 Controls and power system

For propulsion efficiency studies, the vehicle does not possess navigational instrumentation. The testing consisted of commanded tail movements, and not of specified vehicle trajectories. In order to accomplish this, feedback control was implemented from the tail in order to determine its deflection. The actual control law itself was quite simple at this point, and was implemented on a simple MC68HC11 microprocessor. The initial control law for each set of wires is a phased full-voltage command, in order to actuate the wires sequentially along the length of the fish and achieve the travelling waveform. Eight binary output channels were required to control each set of wires, and up to eight A/D input channels. Eight binary input channels were sufficient as only the phase transition information was needed. The output channels controlled FETs for the individual wires.

4.2 Kinematics of the spline motion

The static model of the fish was defined using the variables shown in Figure 4, all defined in the tail coordinate system. The spline is divided into spline segments by vertical lines, and circles represent the pivot locations.

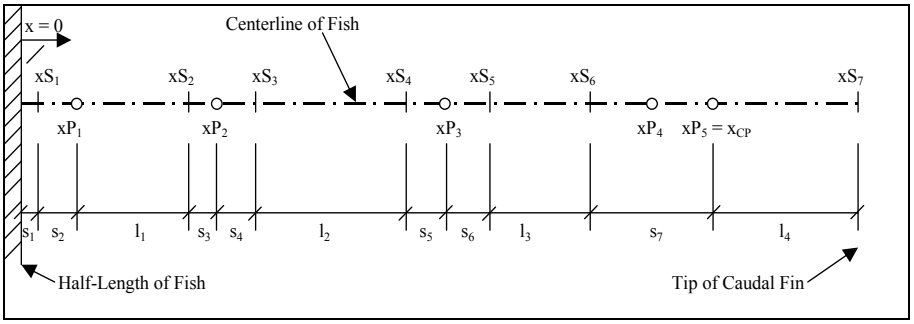


Fig. 4. Relevant coordinates of the tail

The deformed locations of the endpoints of the splines are labelled (x_i,y_i) and the deformed pivot locations (xp_i,yp_i) as illustrated in Figures 5 and 6.

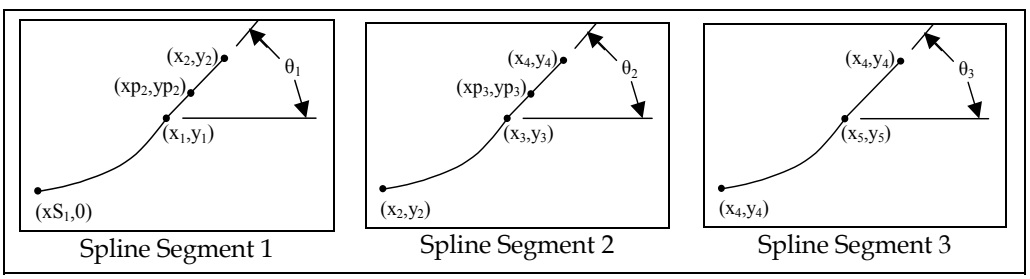


Fig. 5. Deformed spline coordinates

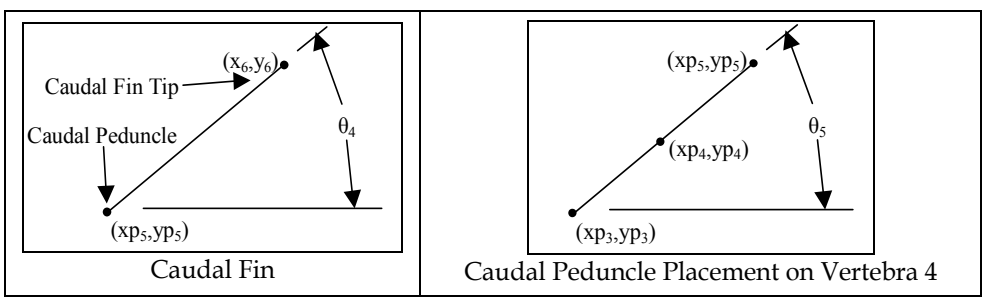


Fig. 6. Deformed Caudal Fin Coordinates

In order to determine a physically realistic pivot location, the length of the splines must be maintained, assuming there is no in-plane straining. The arc-length formula is used for this:

$$l_s = \int_{x_s}^{x_i} \sqrt{1+\left(\frac{dy}{dx}\right)^2}\, dx \tag{1}$$

where x_s is the coordinate of the starting point of the spline, x_i is the endpoint of the spline in question and y is the defining function of the spline in question. The slope of the travelling wave is also needed, as follows:

$$TW'(x) = 2A_0\, x \sin(\omega t - kx) - A_0\, k\, x^2 \cos(\omega t - kx) \tag{2}$$

Matlab was used to solve the necessary equations, using the built in spline functions. The algorithm implemented was to minimize:

$$\Delta = \left| \theta_i - \tan^{-1}\left(TW'(x)\right) \right| \tag{3}$$

where θ_i is the end slope of the spline in question and the design variable for the optimisation problem. For each unique value of θ_i, there is a unique solution to the placement of the pivot point, since the end slope thereby defined; this was implemented in Matlab by searching for the zero of $\delta = l_i - l_s$ where l_i is undeformed length of the spline in question and l_s is as defined in equation 1. For this optimisation problem, xp_i is the design variable. Once the spline endpoints are determined, the location of the caudal peduncle (xp_5,yp_5) is determined from geometry, followed by the location of the tip of the caudal fin (xp_6,yp_6), using:

$$\begin{aligned} {l_4}^2 &= \left(x_6 - xp_5\right)^2 + \left(y_6 - yp_5\right)^2 \\ y_6 &= TW(x_6) \end{aligned} \tag{4}$$

4.3 SMA strains

Once the endpoints of the rigid Delrin blocks are determined (same as the locations of the endpoints of the splines), the dynamics locations of the ends of the SMA wires can be determined from simple geometry. The actual length l of the wires at all time steps is easily determined (from a knowledge of the location and orientation of the Delrin blocks), and compared to the undeformed length of the wire L. When the tail is straight, there is a pre-strain ε_p in each wire at length L'. Eq. (5) therefore describes the instantaneous strain in the wire and it is independent of L since there is no way to calculate it from the tail parameters, whereas L' can be computed from the geometry of the tail:

$$\varepsilon = \frac{l\left(\varepsilon_p + 1\right) - L'}{L'} \tag{5}$$

The 3-D model of the fish was created in Matlab using surfaces defined by X, Y, and Z matrices. The x-axis is along the length of the fish with the origin at the nose, y is the horizontal plane and z in the vertical. The profiles of the fish body are defined by the same coordinates used in the CAD model of a real tuna and NACA airfoil, and the caudal fin has a simple airfoil cross-section. The cross-sections of the body are ellipses defined by:

$$\begin{aligned} y &= a\cos(\alpha) \\ z &= b\sin(\alpha) \\ \alpha &= 0..2\pi \\ a &= F_{Tuna\_\mathrm{Pr}ofile}(x) \\ b &= F_{Airfoil\_\mathrm{Pr}ofile}(x) \end{aligned} \tag{6}$$

The sections on the tail and nose cone are equally spaced, and the cross-sections for the tail are defined by the rib locations. Using the time history of the spline, the position and

orientation of each rib section and the tail can be determined. Transformation matrices are computed for each section of the moving tail and fin, pre-multiplied together, and then multiplied with the respective coordinates of the cross-sections.

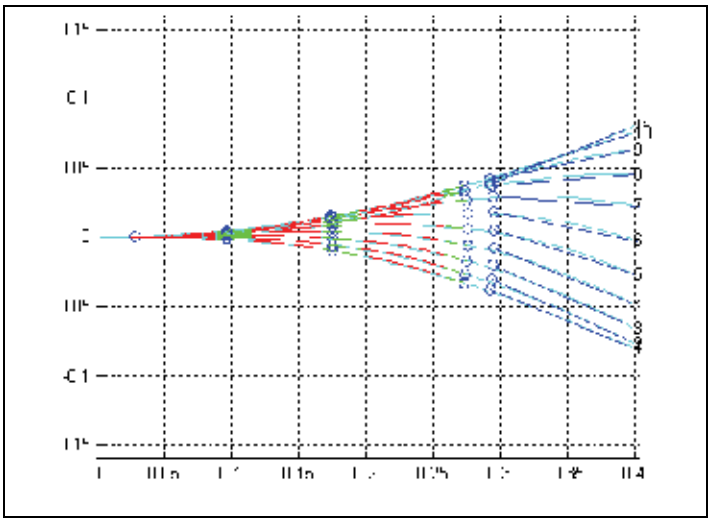


Fig. 7. The tail components superimposed on the travelling wave

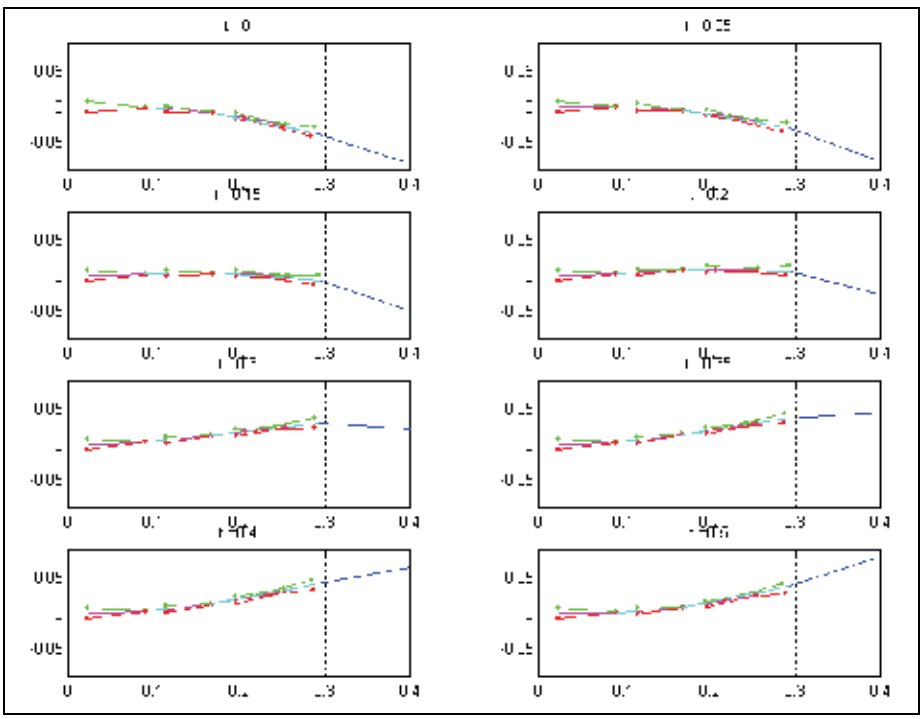


Fig. 8. The time step visualization of the tail motion

There are three main outputs of the spline motion and strain calculations, which can be run for various percentages of the tail period: a superimposed image of the tail position on the

travelling wave, a visualization of the tail at each time step including the SMA wires, and graphs of the strain time histories for both side of the fish, shown in Figures 7 and 8, respectively. The circles in Figure 7 represent the locations of the axles on the tail. All figures are for 0.5s, at a frequency of 1Hz, with A_0 equal to 0.5 (amplitude at tip of tail equal to 8cm), and k equal to 4.

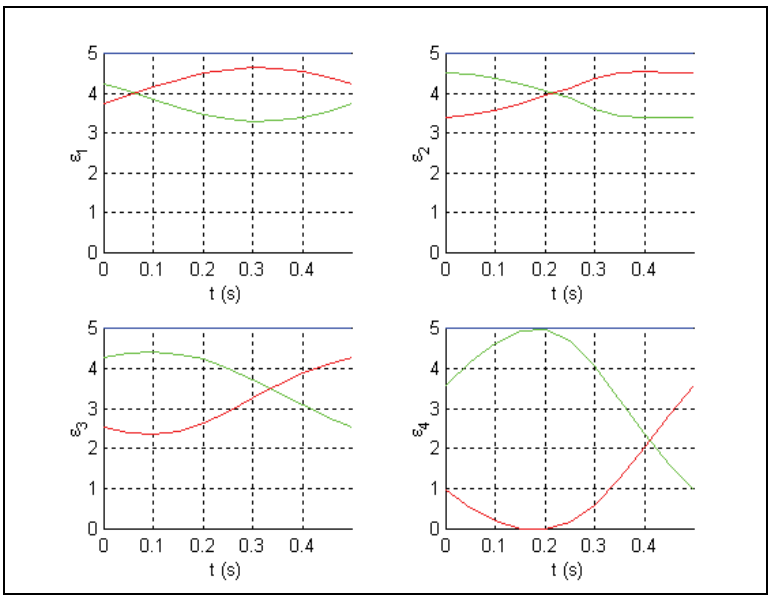


Fig. 9. The SMA wire strain time histories

The ε_1 through ε_4 time histories illustrated in Figure 9 are the strains from the forward most section 1 through aft section 4, for the wires on both sides of the tail. Notice the SMA strains have a phase angle of 180˚ between opposing sides of the same section, and have the profile of sine waves as would be expected. The pre-strains in section one through four are: 0.04; 0.04; 0.035; 0.022.

4.4 The thermomechanical model derivation

In order to size the wires and controller power supply, a thermodynamic model of the heating and cooling of the wires was developed. The lumped capacitance model was used in a 1-D radial formulation, and the accompanying differential equation solved numerically using Matlab. The resistive heating of the wires was modelled based on a supplied current, and free convection was used for the heat transfer at the surface of the wires. The coating on the wires was neglected (since it was assumed to be a very thin film), as was the latent heat of phase transition. The latent heat was initially included in the model, but the small volume of the wires made the factor insignificant. The heat equation for heating of the wire was therefore derived as follows:

$$\dot{E}_g - hA\left(T_s - T_\infty\right) = \rho_{SMA} V c_{SMA} \frac{dT}{dt} \tag{7}$$

With the assumption that $T_s = T$, Eq. (7) was reduced to:

$$I^2 \frac{R_e}{l} - \pi \bar{h} D\left(T - T_\infty\right) = \rho_{SMA} \frac{\pi D^2}{4} c_{SMA} \frac{dT}{dt}$$
$$\Rightarrow \frac{dT}{dt} = 4 \frac{I^2 \frac{R_e}{l} - \pi \bar{h} D\left(T - T_\infty\right)}{\rho_{SMA} \pi D^2 c_{SMA}} \tag{8}$$

In order to calculate the free convective coefficient $\bar{h}$, the Nusselt number was calculated as follows (Incorpera and DeWitt, 1975):

$$\overline{Nu_D} = \frac{\bar{h} D}{k} = \left\{ 0.6 + \frac{0.387 Ra_D^{1/6}}{\left[1 + \left(\frac{0.559}{\mathrm{Pr}}\right)^{9/16}\right]^{8/27}} \right\}^2 \tag{9}$$

with

$$Ra_D = \frac{g\beta\left(T_s - T_\infty\right) D^3}{\nu\alpha} \tag{10}$$

During cooling the internal heat generation term was dropped. The material properties of the SMAs were taken from the product literature for the low temperature wires, and the properties of water and air were implemented in a look-up table for increased accuracy, taken from White (1999).

Performance in air for a variety of wire diameters was first examined to verify the model in relation to the published performance data. While the cooling simulation was quite accurate, the heating simulation required less current than published to attain the required transition temperatures in the wires. The simulation was then repeated using the properties of water, where the high heat transfer coefficients were found to greatly increase the required current when heating, and drastically reduce the cooling times, as reported in the product specifications. Figures 10 and 11 show the simulation results for heating and cooling respectively, along with the finish temperatures for the phase transitions. The numbers in 0 indicate the wire diameter in μm, and both results used an ambient temperature of 10˚C. The results for heating and cooling in water agree with both the product literature and experimentation with a number of test specimens.

Based on this data, the 250μm low temperature wires were chosen for the prototype, and the power supply was sized to deliver up to 3A per wire, since not all wires would be actuated simultaneously.

Since no feedback mechanism was designed into the prototype, an accurate simulation of the mechanical behaviour of the wires was essential. The thermomechanical behaviour of SMAs is only just being to be carefully studied and quantified. The literature commonly refers to a dual kriging model to describe the behaviour of SMAs, relating temperature, strain, and applied stress on a three-dimensional surface, shown in Figure 12.

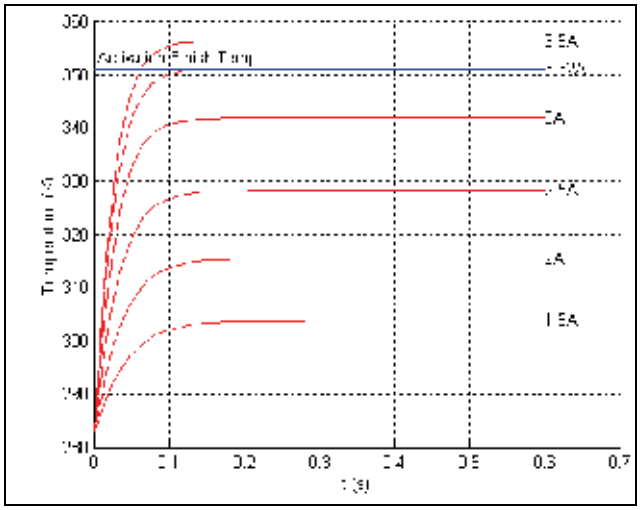


Fig. 10. The Flexinol 250 μm Diameter Wire Heating for Various Current Values

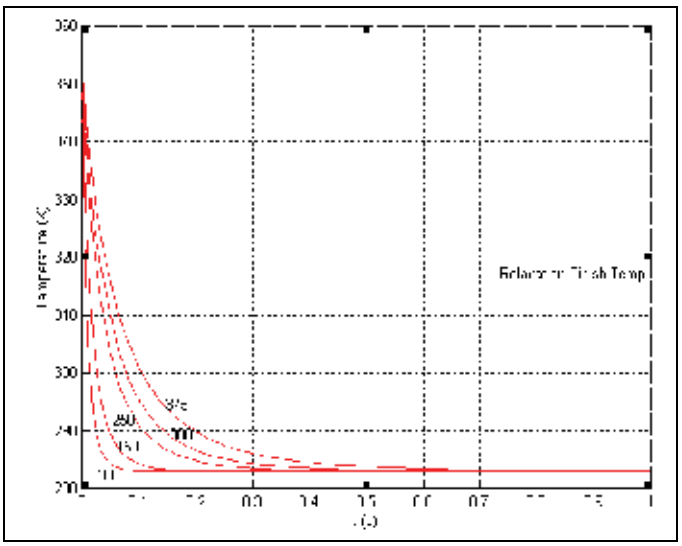


Fig. 11. Flexinol wire cooling in water for diameters of 100, 150, 250, 300 375μm

In order to determine the correct position in that volume at any time, subject to any imposed mechanical or thermal loading, the initial condition of the material must be known. A simpler model of the SMA relates only temperature and strain level, and is adequate for this application. Figure 13 illustrates this relationship, including the transition from martensite to austenite on heating and the reverse phase transition upon cooling and straining. M_s and A_s indicate the temperatures at which the phase transitions are estimated to start, while M_f and A_f denote the finish temperatures of the phase change. As stated in the introduction, in order for the wire to return to its initial strain level, an external biasing force must be applied. Therefore, the wire is assumed to start at an initial pre-strain, point 'A', and shorten to zero strain on heating, point 'B'. While cooling, a force is applied to the wire allowing it to return to its initial strain level along the lower path. The maximum strain

level must be kept below 5% in order to ensure the longevity of the wire. On the fish prototype, the wires are strained back to their initial level by a combination of the set of wires on the opposing side and the spline's bending stress.

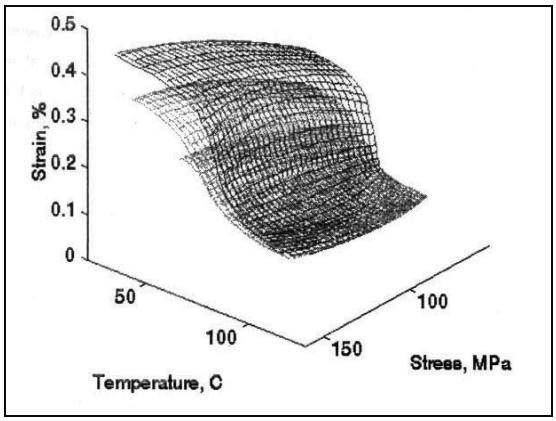


Fig. 12. Heating thermal cycles corresponding to an initial content of 40, 70 and 100% of martensite: the hysteretic volume is created using the sampling sets obtained for loads of 53, 107 and 160 MPa (Volkov, Trochu and Brailosvski, 1999).

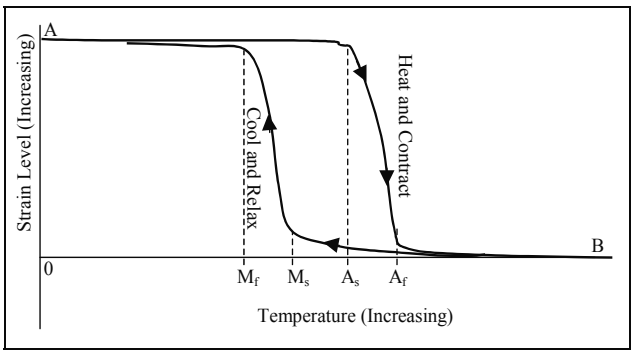


Fig. 13. Graph of Strain Level Versus Temperature for SMA

In order to implement a thermo mechanical model, only the transition from martensite to austenite was considered, and assumed to be linear according to Eq. (11). The reverse transformation could be formulated in an analogous manner.

$$T = \frac{A_s - A_f}{\varepsilon_i}\varepsilon + A_f \tag{11}$$

where ε_i is the strain in the wire at the start of the transition. Notice that this equation defines the required temperature of the wire based on the required strain.

5. Controller design

The length of the wires and corresponding resistances allowed a 5V source to be used to drive enough current through the wires for the anticipated heating requirement. A pulse width modulation scheme was used to control the voltage applied to the wire and the

temperature of the wire, with a 5V maximum. Using the strain time histories from the structural simulations run at 1Hz (tail beat frequency) as the input to the control law, a method was derived to compute the control files used by the ACE controller to move the tail.
The update frequency of the control law was nominally set at 20 Hz, so 20 commands were needed over a 1 second time interval. To avoid damaging the wires, the wires on opposite sides of the tail are never actuated simultaneously. Consequently, the commands were computed for the time periods of decreasing strain. The strain during these time intervals was converted to a commanded temperature using Eq. (11). As an adequate approximation and to avoid complicated analysis, those temperatures were then converted to voltages using the required voltage at steady state conditions. While not strictly true, the solution of the differential heat equation for the controller was not warranted at this stage of development, and so the following equation was derived for the steady state temperature by dropping the transient term.

$$\begin{aligned} \frac{V^2}{R} &= hAl\left(T - T_\infty\right) \\ \Rightarrow V &= \sqrt{\frac{\pi D^2}{4} Rl\left(T - T_\infty\right)} \end{aligned} \tag{12}$$

The actual entries in the control file were then determine by the following equation:

$$PWM = \frac{V}{5}\left(Max\_PWM\_Unit\_Count\right) \tag{13}$$

6. Mast drag prediction

In order to simplify the construction of the tow tank apparatus, the drag of the mast was not isolated from the load cells. Consequently, the drag of the mast must be known in order to determine to thrust of the fish. This value can be estimated both experimentally in the tow tank and by testing of the mast in isolation. Experimental testing introduces other effects, however, such as the vortices shed off the tip of the mast. For this reason, an analytic prediction was sought, using a potential flow panel method combined with boundary layer estimation.
The program DesignFoil (www.designfoil.com) implements a panel method for 2-D thick airfoils combined with boundary layer analysis based on the theory of T. von Kármán and K. Pohlhausen. It also provides a good user interface for airfoil coordinate definition, and outputs coefficients of lift, drag, and pitching moment, given the coordinates of the airfoil and the Reynolds number. The coordinates of the mast were measured, and then fed into the program, interpolated at 200 points on the top and bottom of the section. It was found that the results from the program converged to a steady solution when more than 300 total points were used to define the airfoil. The results of this analysis are shown in Figure 14, with the following constants used for the mast and water:

l = 2.25" (mast chord length), s = 0.56m (submerged length of mast), $F_D = c_d\left(\frac{1}{2}\rho V^2 ls\right)$

ρ = 1000 kg/m^3 , ν = 1.3x10^{-6} m^2/s @ 10°C.

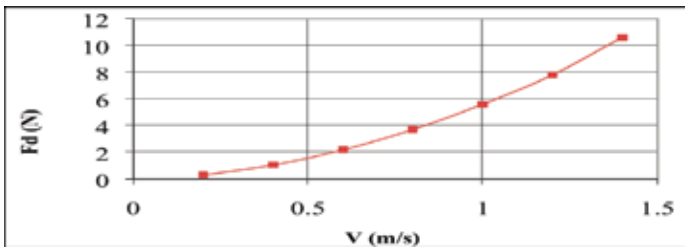


Fig. 14. Predicted mast drag as a function of velocity

7. Prototype I: thrust experiments

The proposed design and the preliminary calculations proved to fulfil the function extremely well, with a minimum of resistance to movement. The SMA actuators have also proven themselves capable of generating a wide range of motion in the tail. Moreover, the controller and circuit design proved effective in providing fine motion control of the SMAs. The structural simulations have demonstrated the ability of the tail to move according the prescribed travelling wave motion, and provided the needed input for the design of the control law. The thermodynamic model derived and implemented appears to agree well with experimental results. Combined with a simple thermomechanical model of the SMA wires, controller files were generated for motion of the tail underwater to be used in the testing regime.

Next, the prototype vehicle was manufactured. Figure 15 shows the prototype in motion in the test tank. Swimming motion was achieved and initial thrust measurements were taken. The complicated heat transfer conditions made smooth activation of the SMA wires difficult. The actuator force was also somewhat binary in nature, as the material passed through the transition temperature. This resulted in an uneven "jerky" motion, especially when tested in air. The damping effects of the water and skin lessened these effects, but the motion was not perfectly fluid from port to starboard.

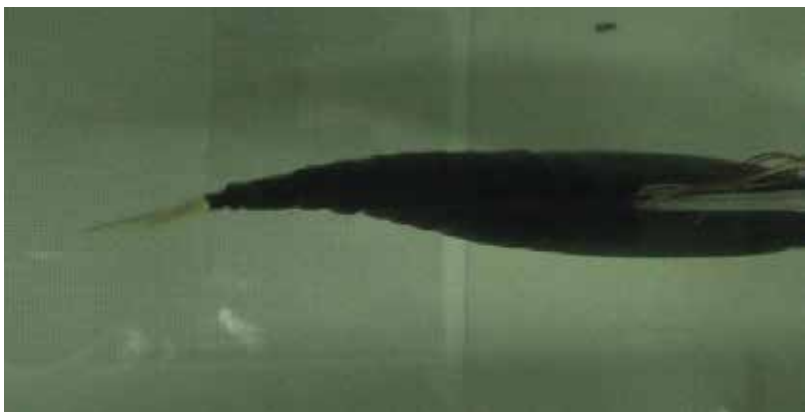


Fig. 15. Prototype Testing in the tank

The speed at which SMA wires can operate as actuators is limited by the rates at which they can be heated and cooled. A further constraint is the software used for the control program,

which has a limited cycle frequency which affects the rate of pulse-wide-modulation (PWM) that can be achieved. Both of these constraints limited the maximum tail beat frequency to 0.5 Hz. Even at this low frequency, the power requirements were measurable.
The current sent to each wire was measured using a digital multi-meter and is presented in Table 2. The current required at each section was different because of the different heat transfer conditions along the length of the tail section, but symmetric about the centreline.

	Port Wires				Starboard Wires			
Vertebra #	1	2	3	4	1	2	3	4
Max current (A)	7.4	10	6.4	3.6	7.4	10	6.4	3.6

Table 2. Current sent to individual wires

The current sent to each wire changes during one period. For this reason it is difficult to accurately calculate the power consumption of the prototype using the basic multi-meter available. Since exact power consumption data was not needed, a more complicated data acquisition system was not implemented. To calculate a rough estimate of power consumption, the current in the two supply wires to the power supplies (instead of individual wires) was measured. The current draw on these wires was found to be relatively steady. Table 3 contains the maximum and minimum current in supply wire one (for vertebrae 1 and 4) and two (for vertebrae 2 and 3). The minimum and maximum power consumptions were calculated to be 292.8 W and 333.6 W respectively.

Current in Supply Wire 1 (Section 1and 4)		Current in Supply Wire 2 (Section 2 and 3)	
Max	Min	Max	Min
15.9 A	13.8 A	11.9 A	10.6 A

Table 3. Maximum and minimum current draws in supply wires 1 and 2

In fish motion, it is of interest to know the amplitude of the wave that the tail section follows, as well as the angle of attack of the caudal tail fin. A digital video camera was used to capture the motion of the fish swimming. Using a 5 mm grid on the bottom of the test tank, the amplitude of the motion and angle of attack of the caudal tail fin was observed as shown in Figure 16.

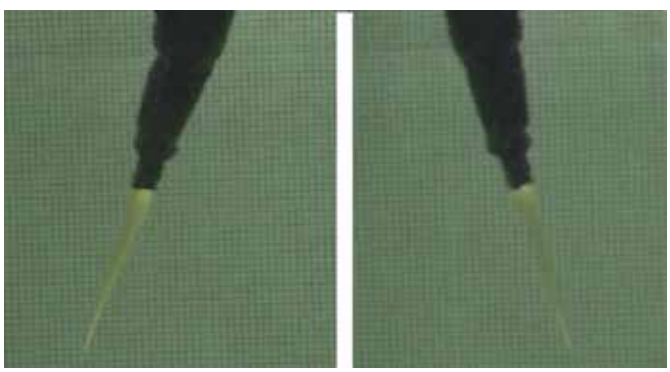


Fig. 16. Maximum displacement and angle of attack of caudal tail fin

Properly installing each SMA wire to the exact length was difficult because of the attachment design. The result was that the amplitude of motion on each side was not perfectly equal. It was predicted that the range of motion of the caudal fin of the prototype would be 10^0 and the amplitude will be 0.08 m. Measurements taken from the video footage gave the results presented in Table 4. Evidently amplitude was over-predicted and angle of attack under-predicted in the design simulations. Note that the camera position was stationary, creating a parallax effect due to the single focal point. This was compensated for in the measurements of amplitude and angle of attack. From measurements taken in the SMA wire calibration process, strain in each wire is estimated to be 5% ± 0.5%. This is the maximum repeatable strain that the SMA can recover from. Using the load cell mounted on the test jig, the forward thrust developed by the prototype was measured. The thrust was found to vary over one period of wave motion, as expected. The maximum force that the prototype generates is 1 N.

Port		Starboard	
Max Tail Amplitude	Max Angle of Attack	Max Tail Amplitude	Max Angle of Attack
5.6 cm	17^0	5cm	14^0

Table 4. Maximum amplitude and angle of attack of caudal tail fin

Given the power consumption during operation, the overall level of performance, particularly the thrust developed, was not satisfactory. There are a number of contributing factors. The first limitation due, to the SMA actuators, is the speed of operation. The maximum operation speed of 0.5 Hz is much lower than other prototypes currently in testing. This is also lower than typical fish non-dimensional tail beat frequencies. The control software is currently the limiting factor on the frequency, and it is believed that an operation speed of 1 Hz (maximum attainable using SMA) would produce much better results.

The second limiting factor on the performance is the amplitude of motion, particularly the displacement of the caudal tail fin. With the SMA wires operating at 5% strain, they do not produce enough displacement for the body of the fish or the caudal tail fin. The potential flow analysis predicted that the optimal angle of attack is 30^0 for maximum thrust. The prototype was only able to produce a maximum angle of attack of 17^0.

Nevertheless, the emulation of the swimming mode of a Bluefin tuna for UUV propulsion presents exciting possibilities for performance improvements over more traditional designs. The vehicle design, using an adaptive structures approach, has been able to realize a significant reduction in the level of complexity of the vehicle. Construction and testing of the SMA fish prototype has highlighted the benefits and challenges inherent in this approach to biomimetics. While the magnitude of thrust generated was not high enough, its low value can be attributed to the control software, rather than mechanical design. Future prototypes may utilize faster control software and a degree of freedom for the entire body to enhance performance. In addition, a tail section using conventional mechanical servo mechanisms for actuation is being developed to better understand the issues associated with fish motion,

independently of unique issues associated with the adaptive structures approach of the SMA fish.
For this first prototype, power consumption was not a major design factor. This is because the main goal was to simply verify that forward motion was attainable. However, in a practical application, SMA actuators require too much power to be useful. The 300 W that the fish required would require a power source similar to a car battery for only one hour of operation. It is not a practical approach for autonomous vehicles. Thus, a new design based on servo-motors is presented next in order to overcome some of the limitations dicussed in the SMA design.

8. The servo tuna: prototype II

Based on the lesson learned from the SMA based propulsion, it was observed that the shape adaptation system needs an actuation system that is reliable, controllable, flexible and energy efficient. It was determined that position control using servomotors are much simpler as the degree of rotation is directly proportional to the input duty cycle. The first servomotor-driven prototype had two joints and two servomotors. A tail (caudal) fin was constructed with the same proportions as a Bluefin Tuna. A waterproof case was constructed for the motors because servomotors are not meant to be operated underwater. For ease of construction, a single watertight case was built to house both servos. The case is a rectangular box, machined out of aluminum. There is a channel for an O-ring and tapped holes, where a plexiglass cover was attached. Directly above the output gears of the servos are two holes to allow the spindles to pass through. A counter bore was above both holes, where an O-ring could create a seal between the case and the spindle. The development of the prototype is chronicled in Figures 17-22.
The servo closest to the caudal fin controlled the rotation of the between the servos and the caudal fin. The servo closest to the nose of the fish controlled the caudal fin by way of linkages. Because the links were located on one side of the apparatus, mechanical interference occurred when the tail flapped toward the opposite side. A problem arose from the connection between the motors and the spindles. This tuna used an injection-molded plastic piece to connect the servo to the spindle. The plastic piece was glued and press fit over the bar, which was the spindle. This union held for the first few trials, but after repeated use, the spindle began to rotate in the plastic piece. As a result, the joint being rotated would not reach the same position as the servomotor, causing the flapping of the tail to meander. A new spindle was designed. Also, the placement of the servomotors were changed to avoid interference.
For the complete model, the prototype II ServoTuna uses four servomotors to move four mechanical joints located on the rear half of a tuna-like body. The design of the single link model was changed to accommodate the two additional servos, and the mechanism that rotated the caudal fin was improved to avoid mechanical interference.
The components of the prototype II were made of aluminum. The principle of having a waterproof case for the servomotors was retained, but each servo had its own case. Figure 21 illustrates the isolation of the servomotors that eliminated the problem of parts interfering with each other, as no single joint could rotate more than 90˚. In order to save time, a few parts were modified only slightly from the original servo fish. The bearings and

journals about which the caudal fin rotated were kept and the pieces used to attach the caudal fin to its servo were modified only slightly. They were shortened to offset the length added by the two extra servos.

To simulate the most lifelike swimming motion, the pivot point for the caudal fin was placed as close to the start of the fin as possible. Bluefin Tuna are quite narrow near the caudal fin. In order to keep the shape of the fish as realistic as possible, the servomotor controlling the fin had to be farther back from the joint. Therefore, a linkage between the servo and the pivot point was necessary. Four ball joints were used to transfer the rotation. The ball joints accommodated changing directions of force and the difference in height between where they attached to the spindle and to the fin. The ball joints were connected with a piece of ready rod. This set-up allows the ball joints to be reused if the distance between them is changed.

The other three joints were identical to each other. A bracket was screwed into the back of the preceding servo case. The bracket clamped the spindle and aligning pin. These two parts made the axis the joint will pivot around. The spindle rotated with the motor. The aligning pin slid within a Delrin bearing. The bearing was fit into a recessed circle in the bottom of the case. The purpose of the aligning pin was to oppose the moment created by the weight of the other joints.

The spindles were designed to fit over the splined output shaft of the servo and transfer the rotation to the joint bracket, outside the servo case. It was decided that the spindle should be one solid piece rather than two pieces joined together. The spindle had to be able to pass through the 0.65 cm hole in the case from only one direction, so the end that fit over the motor shaft could have a larger diameter than 0.64 cm. Because the spindle would be slid in from the interior of the case, the spindle had to be able to slide up far enough to be out of the way when the servomotor was inserted. Once the motor was in place, the spindle could be pushed onto the output shaft. The complete ServoTuna fish with the four servomotors is shown in Figure 23.

The cases were boxes made of aluminum and plexiglass. The center of the aluminum had a shape resembling a spool of thread cut out of it. This recess was where the servomotor was placed. The semicircles at the four corners were to give room for the wires exiting the servo and to allow the motors to be removed easily. Around this cutout was a groove meant for a gasket, which sealed the aluminum to the plexiglass cover. Four screws fastened the cover to the aluminum case. Another hole was drilled through the top of the case. This hole served as an exit point for the servomotor wires. Three wires roughly 7.5 cm in length passed through this 0.3 cm hole and were epoxied in place to create a permanent seal. Inside the servo case, these wires connected with those on the motor. This arrangement provided a reliable seal without permanently attaching the motors to the cases.

At the bottom, the bracket was connected to the aligning pin by clamping it in a circular hole. The slight gap between the two ends of the clamp forced the ends together when the screw was tightened. This type of clamp is an effective method of preventing rotation and vertical motion. Also, the aligning pin did not need additional machining, such as holes or notches and, as such, did not require any special clamps. Creating the clamp on the end of the bracket was more time consuming than simply drilling and tapping a hole into the end of the bracket, but the result was a much stronger grip. The servo cases, which protected the motors from water damage, were 5 cm square and 0.65 cm deep. Due to the complex shapes in the servo cases, they were all made using the CNC milling machine.

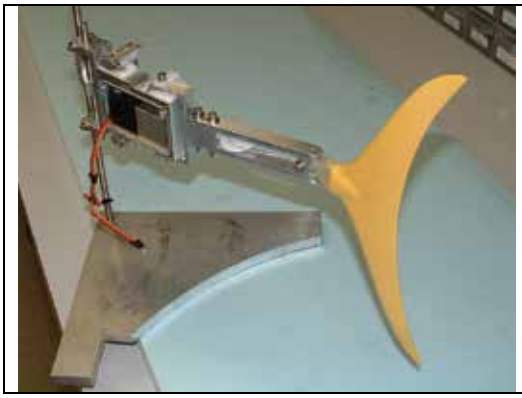


Fig. 17. Single actuator model

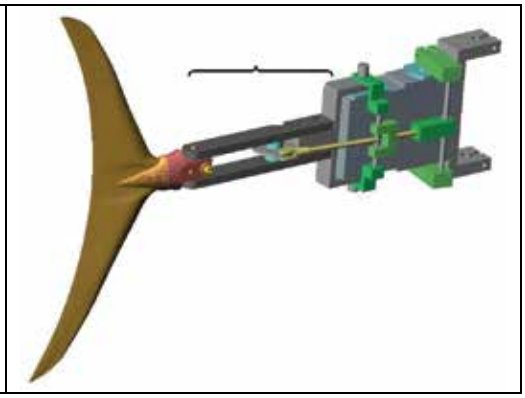


Fig. 18. Linkage Connecting Servo to Caudal Fin Joint

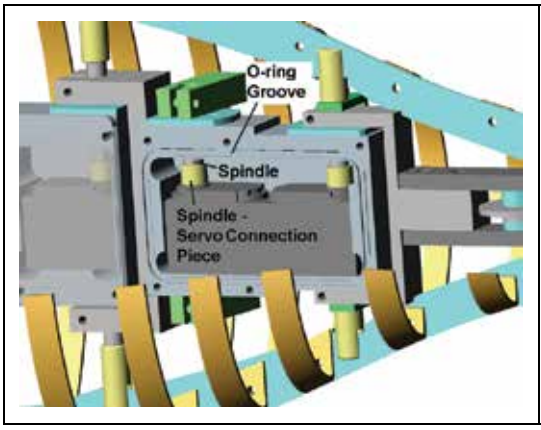


Fig. 19. Servo Holder and Spindles

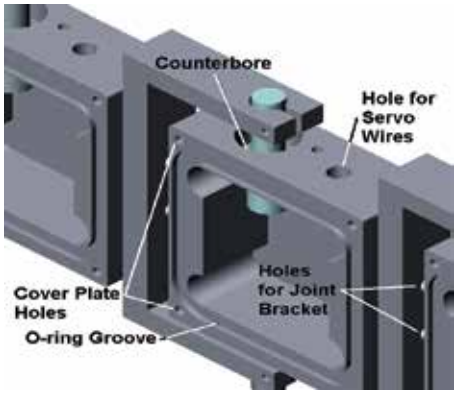


Fig. 20. Servo holder details

Fig. 21. Spindle design details

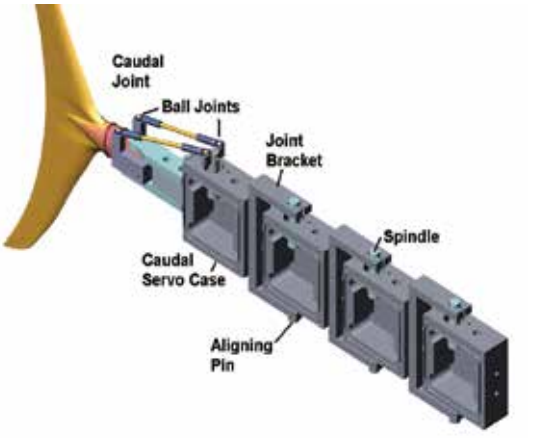


Fig. 22. The four-actuator model

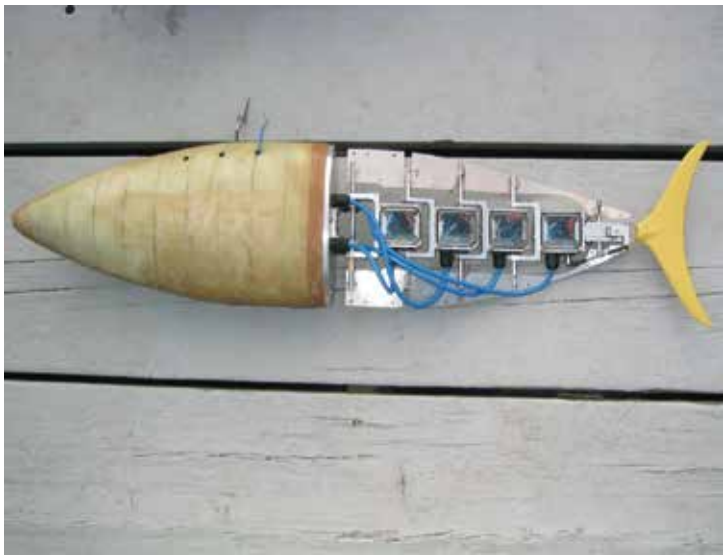


Fig. 23. The Servo Tuna: Prototype II

8.1 Servo controller

There were several tasks the control program had to perform. The most important function was to move the servos in such a way as to create a traveling sine wave along the tail. During experimentation, it was desirable to change the amplitudes of the servos individually and be able to adjust the frequency of the motion.

The servomotors were controlled by Pulse Width Modulation (PWM). The data acquisition cards used with LabVIEW could only support two servomotors. Therefore, to run four servomotors the LabVIEW program would have to use two data acquisition cards. Also, LabVIEW could not use multitasking/multithreading, which allows several processes and functions to operate simultaneously. Multitasking/multithreading would be helpful with controlling four servomotors. The two most viable options were a Motorola microprocessor, the 68HC11 in particular, and a servo controller called the Phidget QuadServo.

The Phidget QuadServo is a small circuit board with plug-ins for four servomotors. It is programmed using Visual Basic, and it plugs into the USB port on any computer. The QuadServo is not a microprocessor because it will not run while disconnected from the computer. This program is object oriented, and, unlike Interactive C, the programmer starts by creating a user interface with various buttons and numerical inputs. A PIC microcontroller was selected for the control system. The model is a PIC16f876, a 24 pin device with PWM capability and a 10 bit A/D built in. The programmer selected is a QuickWriter model from Digikey. The in circuit programming mode was selected so that the robot could be programmed without disassembling it.

A schematic of the controller is shown in Figure 24. The pins from B0 to B7 are used for the servo control. A terminal is used to view the operational menu of the single chip computer. Optionally a palm pilot can be used as the terminal. The max232 chip simply changes the voltage levels from +/- 9 volts on the terminal side to 0 or 5 volt logic on the microcontroller side. A reverse biased diode is used to capture the inductive spikes generated by the motors.

Since they are all in parallel on the power one diode will do the job. Six analog 10 bit inputs are available for navigational control which might use sonar to detect the sides of a pool.

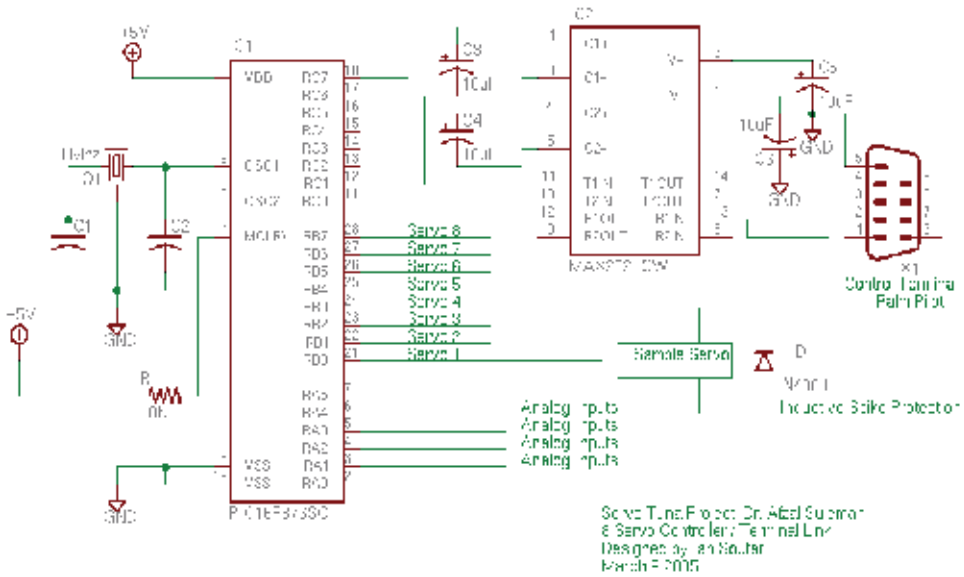


Fig. 24. The servo controller

8.2 The moving wave pattern

The moving wave is achieved by taking the tail motor through its range of motion in a sequential way. Each additional motor starting at the tail has its range of motion cut in half as compared to the previous one. The sine wave passes down the fish with the tail moving the most. The movement tapers off as distance from the tail increases. The code is written for a commercial PIC microcontroller C compiler by Custom Computer Services. The compiler is called PCH.

The fish was prepared for free swimming. A 4 amp hour 6 volt gel-cell type battery was added internally. Surface swimming was effective and achieved a swimming speed of about 0.3 meters/second. The swimming was done outdoors at Prior Lake. In the swimming tests, the most efficient algorithm was the straight S-wave, without the damping factor. The fish was programmed with regular turns which were achieve by simply biasing the tail movements to the left or right. Running in the indoor tank, swimming efficiency was compared after a number of modifications were made to the design. The measured thrust varied betwen 0.5 - 1.0 N. During experiments, we tested the following different designs:

- Different swimming patterns ranging from a strait S pattern to attenuated S patterns known as “travelling wave” patterns.
- Different elastic skin coverings for the tail section.
- Various flexible sheets of material to connect the tail sections.
- A dive plane mechanism to provide for increased lateral stability and to provide for underwater navigation.

8.3 Prototype II: thrust experiments

The most interesting result was that the "traveling wave" pattern programmed into the microcontroller did not perform anywhere near as well as a simple sine wave programmed wave. When viewing the motion of the tail from above in the water the sine wave pattern, due to the forces of the water became a traveling wave pattern. This occurred because the servo motors at the base of the tail were not able to achieve full displacement due to the forces of the water on the tail sections. As we looked into the displacements of each servo we found that more movement was possible as we get closer to the tail. Finally at the tail the servo was achieving a full displacement.

It was remarkable that the classic "traveling wave" pattern that is in all the literature on fish locomotion may be simply a result of the resistance of the water imposing natural limitations on the muscle movement of the fish. This simplifies the software design. This has important implications for the design of robots. It means that it is only necessary to program in a straight "S" or sine swimming pattern into the tail. When the fish is pushed beyond a certain speed the tail motion will become a "traveling wave" pattern instead. When the "traveling wave" pattern was programmed, it resulted in a poor thrust measurement.

The current design can read analog inputs. Thus it can read a sonar input (used in Polaroid land cameras) to see how close a pool edge is. It can do other things like sense light. The PIC16f876 chip currently used in this research project can sense 5 analog inputs in all. The fish might change direction by 90 degrees whenever it "saw" the pool wall coming up. That would be a good first step into auto-navigation. The sensors were tested and found to be effective in detecting an underwater wall.

9. Lessons learned and concluding remarks

The SMA approach offered low thrust (1 N). This was caused by a limitation on speed of recovery time (1 second). The approach by necessity requires excessive power consumption by a factor of 100 or more. This is because large amounts of energy are required to actuate a submerged piece of SMA wire. The power requirement was in excess of 300 watts. An autonomous vehicle is not likely to be achievable using this approach.

The Servo approach on the other hand presented no such limitations. Power consumption was very modest. Five watts of power was sufficient for 4 servo motors which can be supplied by a small battery. Swimming can be easily programmed into a single chip computer making autonomous craft possible. Sensors can be used for auto-navigation. We have achieved autonomous operation with a single chip computer and a gel cell battery. The thrust of this unit ranged from 0.5 N to 1.0 N.

Waterproofing is very difficult to achieve when a rotating shaft bearing is involved. Waterproofing was achieved by combining the use of O-rings with the filling of the engine cavity with silicone grease. There was no leakage because the water could not displace the grease. The O-ring served only to keep the grease and water from mixing and to keep sand out of the mechanism. This is a key process for the successful construction of underwater robots of all types.

A very interesting result was that the "traveling wave" pattern programmed into the microcontroller did not perform anywhere near as well as a straight sine wave programmed wave. When viewing the motion of the tail from above in the water, the sine wave pattern,

became a traveling wave pattern due to the forces of the water. This occurred because the servo motors at the base of the tail were not able to achieve full displacement. As we looked into the displacements of each servo we found that more movement was possible as we get closer to the tail. Finally at the tail, the servo achieved a full displacement. Another interesting conclusion is that the "traveling wave" pattern documented in all the classic literature on fish locomotion may be simply a result of the resistance of the water imposing limitations on the muscle movement of the fish. *It is not a pattern that is created by the fish as much as it is a pattern derived from the interaction of the fish and the water.* When a "traveling wave" pattern was programmed into the controller, the thrust performance was greatly reduced.

The S-pattern program combined with a speed of motion changes to a "traveling wave", provides greater stability when the fish encounters turbulent water. The motors will travel further when they encounter less resistance on one side of the fish, compensating for the reduced pressure of the water. Similarly the motors will travel less when they encounter an increase in water resistance. The result is a fish movement that is quite resistant to turbulent waters.

This has important implications for the design of robots. It means that it is only necessary to program in a straight "S" or sine swimming pattern into the tail. When the fish is pushed beyond a certain speed the tail motion will become a "traveling wave" pattern instead. The fish will be less affected by turbulent water if it operates in the over driven mode. For this reason it is desirable to choose servo motors which will experience attenuation of their full travel by the forces of the water at the maximum desired speed.

Full Navigational Control: Left and right navigation are easy to achieve by simply putting a left or right bias into the servo-motor nearest the body of the fish. This slants the tail left or right. In our free swimming tests this method worked very well. The dive planes should achieve diving and underwater control when combined with a buoyancy controlling mechanism. The current prototype did not have enough free space in its interior to include this type of control.

Ballasting and Stability: We were able to achieve underwater stability by handing a round lead weight from the bottom of the fish. This ensures that the center of gravity is well below the center line of the fish. By adjusting its position forwards and backwards we can adjust the balance of the fish so that it sits horizontally in the water from nose to tail. It is a simple technique to compensate for the performance of the robot as internal components are added.

Predicted Cruising Distance: The battery used offer 4 Ah. The current consumption of the fish is approximately 1.6 A. This means that we can cruise for approximately two hours (without fully discharging and damaging the battery). The speed was about 0.3 meters/ sec. Therefore the cruising distance was about 2 kilometers.

The following work could be achieved in a future development project:

- Scale up the robot by a factor of 200-300 percent.
- Use more powerful servo motors to achieve greater speeds.
- Research has shown that only 3 tail servos are required. This simplifies the design in many ways.
- Use dive planes for underwater navigation.
- Sonar remote control will be added.
- Add a camera for underwater viewing that can store the images for later recovery.

10. Acknowledgments

The contributions to this research by the following undergraduate co-op students is noted: Erin Cooney, Hugh Patterson and Dennis Otwom, and by the research engineer Ian Soutar.

11. References

Anderson, J. M., Kerrebrock, P. A., "*The Vorticity Control Unmanned Undersea Vehicle[VCUUV]: An Autonomous Robot Tuna*", The Charles Stark Draper Laboratory, Inc. Projects Booklet (2000)

Barrett, D., "*MIT Ocean Engineering Testing Tank Biomimetics Project: RoboTuna*", http://web.mit.edu/towtank/www/tuna/robotuna.html, (2000)

Barrett D., Grosenbaugh M., and Triantafyllou, "*The Optimal Control of a Flexible Hull Robotic Undersea Vehicle Propelled by an Oscillating Foil*", Proceedings of the IEEE Symposium on Autonomous Underwater Technology, Monterey, CA, pp. 1-9 (1996)

Blikhan R., and Cheng J.Y., "*Energy Storage by Elastic Mechanisms in the Tail of Large Swimmers – a Re-evaluation*", J/ Theoretical Biology, Vol 168, pp. 315-321 (1994)

Chiu F.C., Wu C.P., and Guo J., "*Simulation on the Undulatory Locomotion of a Flexible Slender Body*", 1st International Symposium on Aqua Bio-Mechanisms, Hawaii, USA, pp. 185-190 (2000)

Chiu, F.C., Guo J., Chen J.G., and Lin Y.H., "*Dynamic Characteristics of a BiomimeticUnderwater Vehicle*", Proceedings of the IEEE Symposium n Underwater Technology, Tokyo, Japan, pp. 172-177 (2002).

Guo J., Chiu F.C., Cheng S.W., and Joeng Y.J., "*Motion Control and Way-point Tracking of a* Biomimetic Underwater Vehicle", Proceedings of the IEEE Symposium n Underwater Technology, Tokyo, Japan, pp. 73-78 (2002).

Harper K.A., Berkemeier M.D., and Grace S., "Modelling the Dynamisc of Spring-Driven Oscillating –foil Propulsion", IEEE Journal of Oceanic Engineering, Vol.. 23, No.3, pp. 258-296 (1998).

Incropera, F., DeWitt, D., "Fundamentals of Heat and Mass Transfer", 4th Edition, John Wiley & Sons, Canada, Eqn. (9.34), pp. 502

Kumph, J. M., "The MIT Robot Pike Project", http://www.mit.edu/afs/ athena/org/t/towtank/OldFiles/www/pike/index.html, (2000)

Lighthill, M.J., "Aquatic Animal Propulsion of High Hydromechanical Efficiency", Journal Fluid Mechanics, 1970, vol. 44, pp. 265-301 (1970).

Rediniotis, O. K., Lagoudas, D. C., "Theoretical and Experimental Investigations of an Active Hydrofoil with SMA Actuators", Aerospace Engineering Department, Texas A&M University, College Station, Texas 77845-3141

Sfakiotakis M., Lane D.M., and Davies J.B., "*Review of Fish Swimming Modes for Aquatic Locomotion*", IEEE Journal of Oceanic Engineering, Vol. 24, No.2, pp. 237-252 (1999).

Volkov, O., Trouchu, F., Brailovski, V., "*Material Law for NiTi Shape Memory Alloys Based on Dual Kriging Interpolation*", J. of Mechanical Behaviour of Materials, Vol. 10, No. 4 (1999).

Wardle C.S., and Reid A., "*The Application of Large Amplitude Elongated Body Theory to Measure Simming Power in Fish*", Fisheries Mathematics, ed. J. H. Steele, Academic Press, London (1997).

25

Decentralized Control System Simulation for Autonomous Underwater Vehicles

Nanang Syahroni[1], Young Bong Seo[2] and Jae Weon Choi[2]
[1]*Electronics Engineering Polytechnic Institute of Surabaya*
[2]*School of Mechanical Engineering, Pusan National University*
[1]*Indonesia*
[2]*South Korea*

1. Introduction

The problem of power and communication limitation in underwater environment makes it more challenge to increase the degree of autonomy and intelligence for an autonomous underwater vehicle (AUV). An infrastructure of autonomous teleoperation platform for AUV is established and described, which allows control to be shared between the intelligent decision system in AUV system and operators throughout a mission.
In the (Paunicka J.L. et al., 2001, Wills L. et al., 2000, and Wills L. et al., 2003), the information-centric control and engineering have a remarkably successful history of enabling for designing, testing, and transitioning embedded software to unmanned air vehicle (UAV) platforms. A new software infrastructure called Open Control Platform (OCP) will accommodate in changing navigation information and control components, interoperate in heterogeneous environments, and maintain viability in unpredictable and changing environments. The OCP extends new advances in real-time middleware technology, which allows distributed hetero-geneous components to communicate asynchronously in real-time via CORBA middleware. It uses event-based distributed communication and it capable of transmitting events at different priorities. This enables highly decoupled interaction between the different components of the system, which tends to localize architectural or configuration changes that promising to be implemented quickly and high reliability in the real system.
There are many examples of nice control algorithms for AUV which had done in several platforms (Valavanis K.P. et al., 1997), but in the implementation of those control systems in the sense of tightly coupling model in remote operation is widely open for sub-discipline of software engineering. We further investigate how the real-time control system performance could be reconfigured easily both in semi-automatically or manually interventions by remote station, and also develop a simulation platform to support a tuning mechanism of control parameters during runtime (i.e. feedback gains or trajectories) by using Matlab on separated machines connected via CORBA event-channel.
In this paper we organized as follows: Section 2 presents AUV dynamic model, physical values, and control algorithm. Section 3 gives the simulation systems design; include the hardware of simulation workstation, tools and interfaces, and middleware infrastructure. Section 4 presents results from the simulations together with the assumptions of problems solution. The last section covers conclusions.

2. Equations of motion

2.1 AUV dynamic model

The AUV Model for depth control is depicted in figure 1.

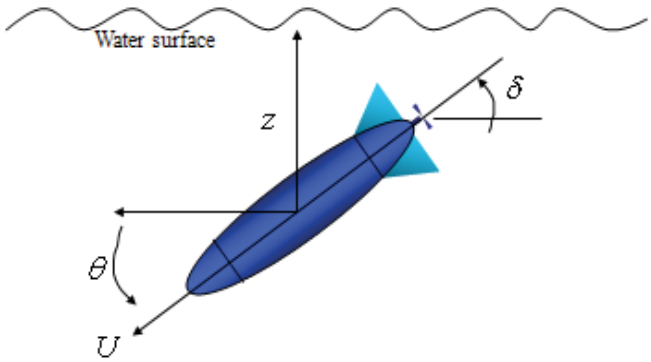


Fig. 1. AUV Model

The simple's form of equation of motion is obtained with body axes coincident with the principles axes of inertia, and the origin at the center of mass center of gravity (CG), for this case the equation in the dimensionless form as in (Sname 1950) are:

$$\begin{aligned} X &= m[\dot{u} + qw - \upsilon r] \\ Y &= m[\dot{\upsilon} + ru - pw] \\ Z &= m[\dot{w} + pv - qu] \\ K &= I_x\dot{p} + (I_z - I_y)qr \\ M &= I_y\dot{q} + (I_x - I_z)rp \\ N &= I_z\dot{r} + (I_y - I_x)pq \end{aligned} \tag{1}$$

The 6DOF components of the rigid body dynamic equations of motion of the submerged vehicle are:

$$\begin{aligned} X &= m[\dot{u} - \upsilon r + wq - x_G(q^2 + r^2) + y_G(pq - \dot{r}) + z_G(pr + \dot{q})] \\ Y &= m[\dot{\upsilon} + ur - wq + x_G(pq + \dot{r}) - y_G(p^2 + r^2) + z_G(qr - \dot{p})] \\ Z &= m[\dot{w} - uq + \upsilon p + x_G(pr - \dot{q}) + y_G(qr + \dot{p}) - z_G(p^2 + q^2)] \\ K &= I_x\dot{p} + (I_z - I_y)qr + I_{xy}(pr - \dot{q}) - I_{yz}(q^2 - r^2) - I_{xz}(pq + \dot{r}) + m[y_G(\dot{w} - uq + \upsilon p) - z_G(\dot{\upsilon} + ur - wp)] \\ M &= I_y\dot{q} + (I_x - I_z)pr - I_{xy}(qr + \dot{p}) - I_{yz}(pq - \dot{r}) - I_{xz}(p^2 - r^2) + m[x_G(\dot{w} - uq + \upsilon p) - z_G(\dot{u} - \upsilon r + wq)] \\ N &= I_z\dot{r} + (I_y - I_x)pq - I_{xy}(p^2 - q^2) - I_{yz}(pr + \dot{q}) + I_{xz}(qr - \dot{p}) + m[x_G(\dot{\upsilon} + ur - wp) - y_G(\dot{u} - \upsilon r - wq)] \end{aligned} \tag{2}$$

where, X, Y, and Z are surge, sway, and heave force; K, M, and N are roll, pitch, and yaw moment; p, q, and r are roll, pitch, and yaw rate; u, v, and w are surge, sway, and heave velocity; x, y, and z are body fixed axes in positive forward, positive starboard, and positive down; I_x, I_y, and I_z is vehicle mass moment of inertia around the x-axis, around the y-axis, and around the z-axis; x_G, y_G, and z_G are longitudinal position, athwart position, and vertical position of center of gravity; ϕ, θ, and ψ are roll, pitch, and yaw angle.

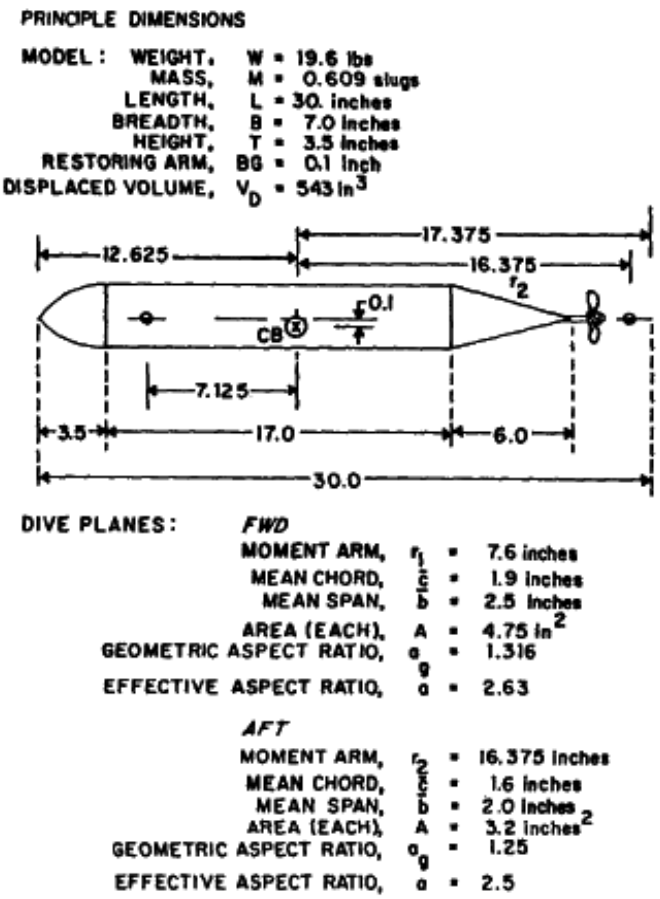


Fig. 2. Physical Dimensions of Vehicle

We can further simply equations (2) by assuming that y_G is small compared to the other terms. After several steps of linearization as in (Cristi, R. et al., 1990, and Riedel J.S., 1993), vertical motion equations become:

$$\begin{aligned} \dot{\theta} &= q \\ (m - Z_{\dot{w}})\dot{w} - (mx_G + Z_{\dot{q}})\dot{q} &= Z_w U w + (m + Z_q) U q + U^2 Z_\delta \delta \\ (-M_{\dot{w}} - mx_G)\dot{w} + (I_y - M_{\dot{q}})\dot{q} &= -(z_G W - z_B)\theta + M_w U w + (M_q - mx_G) U q - M_\delta U^2 \delta \\ \dot{z} &= -U\theta + w \end{aligned} \tag{3}$$

It can rewritten in the matrix form

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (m - Z_{\dot{w}}) & -(mx_G + Z_{\dot{q}}) & 0 \\ 0 & (-mx_G - M_{\dot{w}}) & (I_y - M_{\dot{q}}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{w} \\ \dot{q} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & Z_w U & (m + Z_q) U & 0 \\ -(z_G W - z_B B) & M_w U & (M_q - mx_G) U & 0 \\ -U & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ w \\ q \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ Z_\delta U^2 \\ M_\delta U^2 \\ 0 \end{bmatrix} \delta \tag{4}$$

Then, an AUV dynamic equation typically represented using the notation: $\dot{x} = Ax + Bu$, with state vector $x = [\theta \quad w \quad q \quad z]^T$ and the control input $u = \delta$:

$$\begin{bmatrix} \dot{\theta} \\ \dot{w} \\ \dot{q} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ a_{21} z_{GB} & a_{22} U & a_{23} U & 0 \\ a_{31} z_{GB} & a_{32} U & a_{33} U & 0 \\ -U & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ w \\ q \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ b_1 U^2 \\ b_2 U^2 \\ 0 \end{bmatrix} \delta \tag{5}$$

where:

$$Dv = (m - Z_{\dot{w}})(I_y - M_{\dot{q}}) - (mx_G + Z_{\dot{q}})(mx_G + M_{\dot{w}})$$

$$z_{GB} = z_G - z_B$$

$$a_{21} = -\frac{(mx_G + Z_{\dot{q}})W}{Dv}$$

$$a_{22} = \frac{(I_y - M_{\dot{q}})Z_w + (mx_G + Z_{\dot{q}})M_w}{Dv}$$

$$a_{23} = \frac{(I_y - M_{\dot{q}})(m + Z_q) + (mx_G + Z_{\dot{q}})(M_q - mx_G)}{Dv}$$

$$a_{31} = \frac{-(mx_G - Z_{\dot{q}})W}{Dv}$$

$$a_{32} = \frac{(m - Z_{\dot{w}})M_w + (mx_G + M_{\dot{w}})M_\delta}{Dv}$$

$$a_{33} = \frac{(m - Z_{\dot{w}})(Mq - mx_G) + (mx_G + M_{\dot{w}})(m - Z_q)}{Dv}$$

$$b_1 = \frac{(I_y - M_{\dot{q}})Z_\delta + (mx_G + Z_{\dot{q}})M_\delta}{Dv}$$

$$b_2 = \frac{(I_y - Z_{\dot{w}})M_\delta + (mx_G + M_{\dot{q}})Z_\delta}{Dv}$$

Finally, we refer to the physical parameter of NPS AUV1 in the (Healey A.J. et al., 1997) as depicted in figure 2, the state space became:

$$\begin{bmatrix} \dot{\theta} \\ \dot{w} \\ \dot{q} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0.0175 & -1.273 & -3.559 & 0 \\ -0.052 & 1.273 & -2.661 & 0 \\ -5 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ w \\ q \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0.085 \\ 21.79 \\ 0 \end{bmatrix} \delta \tag{6}$$

2.2 LQR formulation

The general problem of Linear quadratic (LQ) optimal control problem to find a control law $u(t) = -Kx(t)$ to minimize $J = \int_0^\infty \left[x^T(t) Q x(t) + u^T(t) R u(t) \right] dt$, where $K = R^{-1} B^T P$ and P is solution of Riccati equation $A^T P + PA + PBR^{-1} B^T P + Q = 0$, and weighting matrices are: $Q \geq 0$ and $R > 0$.

Linear quadratic (LQ) servo is command following regarding to the reference input. An *n*th-order system having *r* inputs and *m* outputs: $x(t) = \left[x_r(t); y_p(t) \right]$, where $y_p(t) \in R^{m\times 1}$ is output and $x_r(t) \in R^{(n-m)\times 1}$ is rest of system state, $e(t) = r(t) - y(t) \in R^{m\times 1}$ is the error vector with $r(t) \in R^{m\times 1}$ is plan output and $y(t) \in R^{m\times 1}$ is reference. Consider state space model:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{7}$$

with $y(t) = C_p x(t)$, $x_r(t) = D_p x(t)$, and C_p is $\left[0_{m\times(n-m)} \quad I_{m\times m} \right]$, D_p is $\left[I_{(n-m)\times(n-m)} \quad 0_{(n-m)\times m} \right]$. Consider control input $u(t) = -Gx(t)$, and control gain G consists of $G = \left[G_y \quad G_r \right]$, where the gain vector is $G_y \in R^{m\times m}$ and $G_r \in R^{m\times(n-m)}$. The control law than became:

$$u(t) = -G_y y(t) - G_r x_r(t) + G_y r(t) \tag{8}$$

substitute (8) into (7), we have differential equation of close-loop system as follows:

$$\dot{x}(t) = \left[A_r - BG_y C_p - BG_y D_p \right] x(t) + BG_y r(t) \tag{9}$$

Solution of equation (9) could be solving by using Runga-Kutta 4th order approximation:

$$x(t+1) = x(t) + \frac{(k1 + 2k2 + 2k3 + k4)}{6} \tag{10}$$

where:

$$\begin{aligned} k1 &= f(t, x(t)) \\ k2 &= f(t + \frac{h}{2}, x(t) + \frac{h}{2} k1) \\ k3 &= f(t + \frac{h}{2}, x(t) + \frac{h}{2} k2) \\ k4 &= f(t + h, x(t) + hk3) \end{aligned}$$

2.3 Q, R Selection via GA

In order to choose the weighting matrices of LQR control algorithm, it is often by trial and error method. Another method to find an expected values of weighting matrices component is using guided random search, this is one of the easiest way and it could be handle by search function in Matlab simulation source. The most simple and well known technique for guided random search in machine learning application is using the Genetic Algorithm (GA), as shown in figure 3.

Genetic algorithm performed chromosome operation and individual evaluation. In the first iteration, chromosome operation generates the initial population consists of 10 populations randomly with logarithmic distribution. Each population consists of 5 chromosomes, 4 chromosomes of *Q* vector matrix, and 1 chromosome of *R* matrix. Individual evaluation consists of selection procedure according to the fitness function variables of time domain characteristic; such as maximum overshoot tolerance is 10%, maximum rise time tolerance is

15%, maximum settling time tolerance is 10%, maximum steady-state error tolerance is 0.2%, also to check the location of the close-loop poles whether it near by imaginary axis. If there are several values are passed through those selection criterions than GA choose 2 best values, then it transfer to become a new candidate in next generation selection together with new 8 populations, where 6 from randomly generated and the 2 other new populations from both crossover and mutation from 2 best previous populations. Finally, in the end of iteration only one best value is selected.

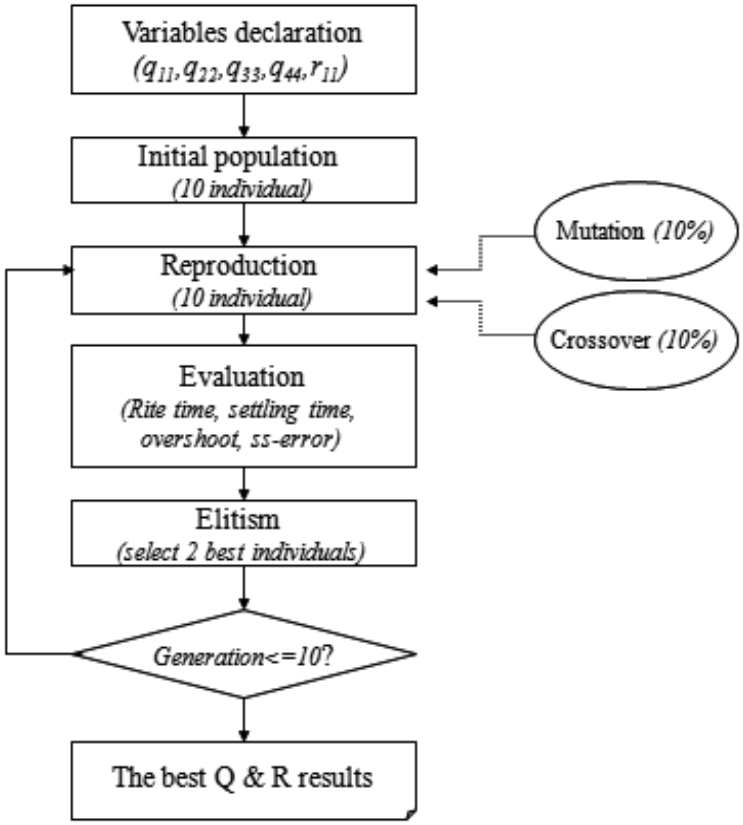


Fig. 3. GA Flowchart

An Implementation of GA into LQ Servo control algorithm as depicted in figure 4. The purpose of this combination is to obtain the global optimal feedback gain *K*, which could be change anytime during runtime. The purpose of changing *K* is to suppress undesired controller performances.

3. System environment

In this paper, an AUV decentralized control system approach is investigated, this will open a possibility to enable control system component to interact between various control components on the simulation network infrastructure. During the course of theoretical studies to simulation platform development, an OCP has emerges some promises to overcome any boundary for both in control system domain and network infrastructure domain.

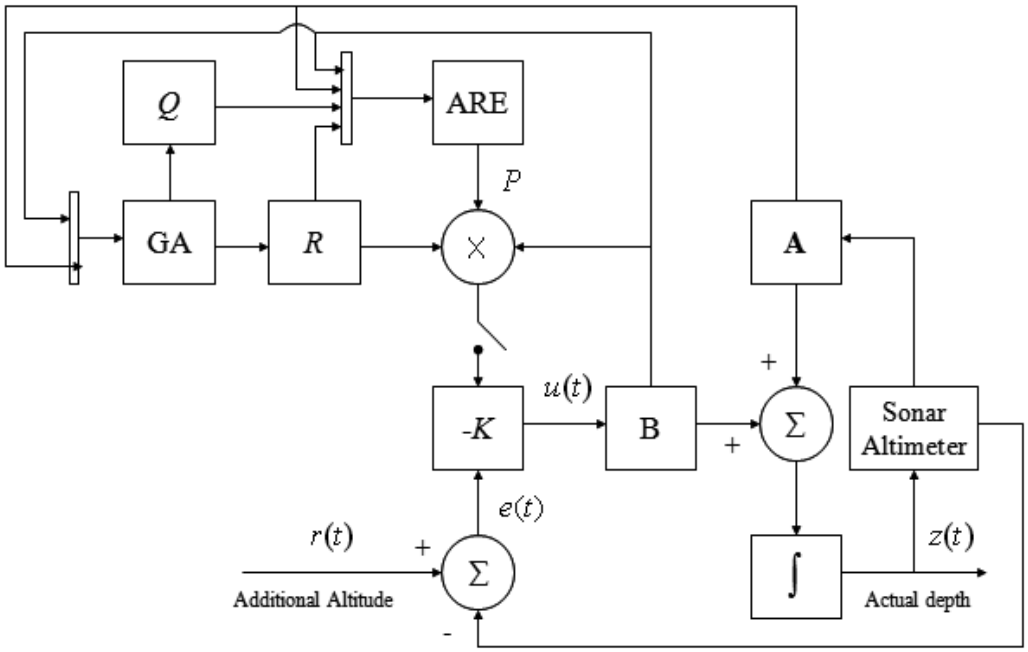


Fig. 4. LQ Servo with GA

3.1 Simulation workstation

To implement the proposed AUV depth control system simulation, information may needs to be rerouted between AUV subsystems or control components. In this situation, sometimes a certain data may became temporarily very important and at other time not needed at all. In figure 5, a simulation system consist of two node connected with a general 10 Mb Ethernet, PC1 as a server and PC2 as client, every node will consists of two blocks, first block consist of vehicle model and control algorithm, and second block is consist real-time network components that support hard control reconfiguration.

The two PCs as server and client are connected via middleware communication using TAO-CORBA Event channel, which is described more details in (Schmidt D. C. et al., 2000). A PC1 running a Matlab simulation of vehicle model and control algorithm, while PC2 running a Matlab simulation as sensor source and mission control station to allows a user to dynamically modify any parameters during a runtime simulation.

3.2 Simulation Tools and Interfaces

Generally, in the development step, mostly control engineer test the new control algorithm in Matlab environment. Matlab is a convenient tool for graphical plotting; it is relatively difficult to use C++ to plot system responses in multiple dimensions. However, C++ is widely used in real-time data acquisition and control in industrial applications.

The interface between C++ and Matlab offers a significant improvement in data acquisition and control system analysis. This makes the analysis for complicated systems possible in the real world. Using the interface method, it is much more convenient to perform matrix operations with real-time controllability.

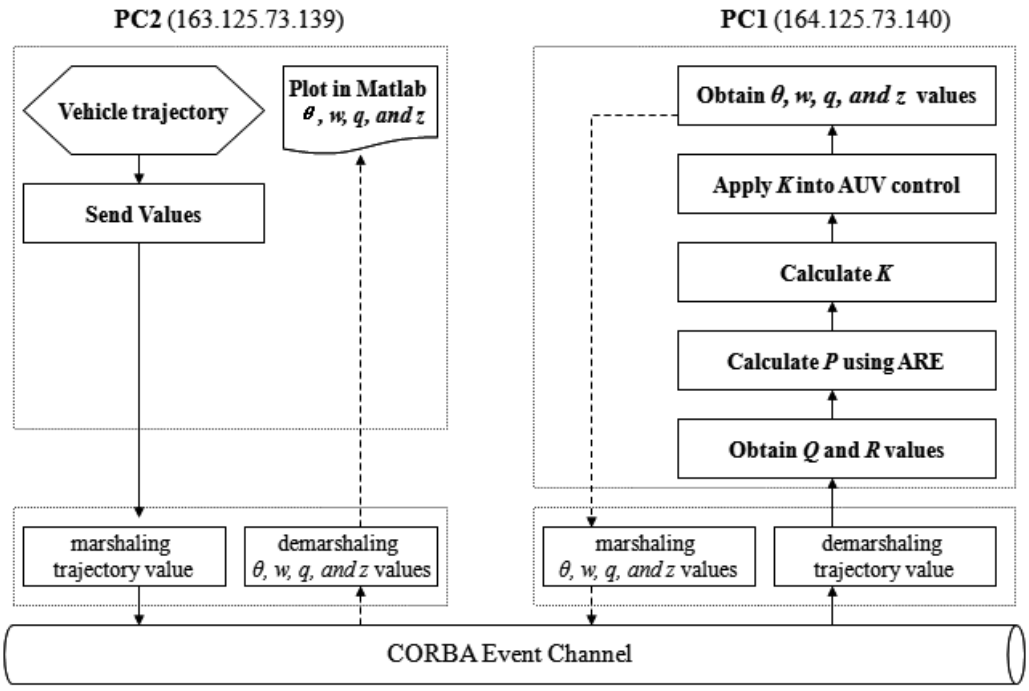


Fig. 5. AUV Simulation Block Diagram

Another point is, for real-time systems, especially for the multiple variables control system, a state variable matrix has to be used to make the real-time analysis based on the state feedback from the system outputs. By using the interface between C++ and Matlab, a lot of data analysis and real-time control tasks for actual systems are possible. Another advantage to using the interface between C++ and Matlab is to handle multi dimension matrix operations and continuous plotting of system responses. Almost all data acquisition and control processes need time response of trajectory in real-time.

In figure 6, illustrates the block diagram of operation principle C++ and Matlab interface. Via this interface, C++ program should collect the data from PC2 through CORBA and create the data variables in ASCII format. Matlab first picks up the data from the data variables have stored by C++ and then performs the matrix operations based on the data. The results can be sent back to C++ by Matlab in the ASCII variables, while C++ program continue to executing the communication task to send a Matlab results.

The interface between Matlab and C++ in order to transfer a data through CORBA event channel is not so complicated, although for control engineers, this method offers a significant improvement in data acquisition and control system analysis; this makes the analysis for complicated systems possible in the real world.

3.3 Middleware infrastructure

The CORBA middleware is an application framework that provides interoperability between objects, built-in different languages, running on different machines in heterogeneous distributed environments. Using a CORBA, a client can transparently invoke a method on a server object, which can be on the same machine or across a network. The

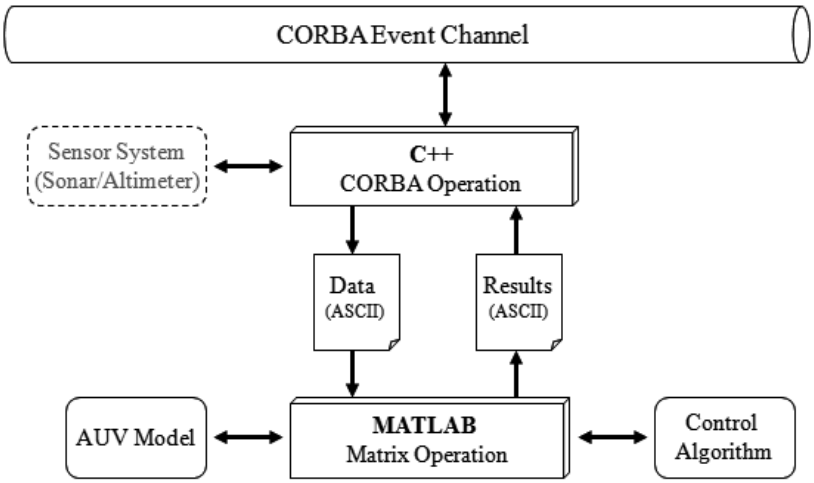


Fig. 6. Matlab and C++ Interfaces

ORB intercepts the call and is responsible for finding an object that can implement the request, pass it the parameters, invoke its method, and return the results.
The CORBA event service provides support for decoupled communications between objects. It allows suppliers to send messages to one or more consumers with a single call. The event service acts as a mediator that decouples suppliers from consumers.

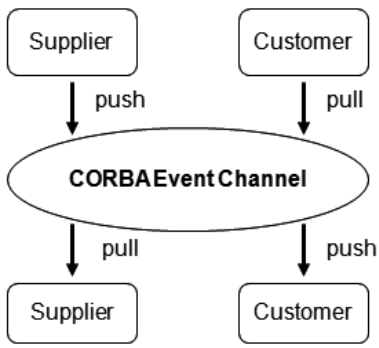


Fig. 7. Participants in the Event Channel Architecture

In figure 7, a CORBA event service provides a flexible model for asynchronous and group communication among distributed and collocated objects. Consumers are the ultimate targets of events generated by suppliers. Suppliers and consumers can both play active and passive roles. An active push supplier pushes an event to a passive push consumer. Likewise, a passive pull supplier waits for an active pull consumer to pull an event from it. Suppliers use event channels to push data to consumers. Likewise, consumers can explicitly pull data from suppliers. The push and pull semantics of event propagation help to free consumers and suppliers from the overly restrictive synchronous semantics of the standard CORBA two way communication model.
In this paper, a CORBA event service implementation focuses on real-time enhancements to the push model, which allows suppliers of events to initiate the transfer of event data to consumers. Suppliers push events to the event channel, which in turn pushes the events to consumers.

4. Simulation results

4.1 Simulation condition 1

Let consider a first example as depicted in figure 8; in this simulation we want to control θ near zero and z near -2 meter with 30 times counter duration. We use a reasonable amount of dive planes to do the job. Assumption: 4° dive planes when pitch angle deviates to 5° from zero, the AUV reaches a depth of -2 meter with 0.32 meter deviation. Therefore, we assume all terms in $Q \rightarrow 0$ and $R \rightarrow 0$, except: $q_{11} = (4/57.2958)^{-2}$=205.21, $q_{44} = (5/57.2958)^{-2}$= 131.31, and $r_{11}= (0.32)^{-2}$=9.76, simulation result as illustrated in figure 8 using solid line. To overcome undershoot and overshoot in the runtime simulation, after duration of $t = 4$ seconds PC2 send new weighting matrices to the PC1 to change a pitch angle deviates to 5° with 0.02 meter deviation at -2 meter depth: $q_{11}=(4/57.2958)^{-2}$= 205.21, $q_{44} = (5/57.2958)^{-2}$=131.31, $r_{11}= (0.02)^{-2}$=2500, simulation result as illustrated in figure 8 using dash line.

The time response of both controller are equal for $t \leq 4$ seconds because of all parameters are same, by intervention from PC2 to PC1 when $t \geq 4$ seconds, the new control parameters are apply during runtime, then it could be seen that time response is improved significantly, especially to suppress undershoot and overshoot.

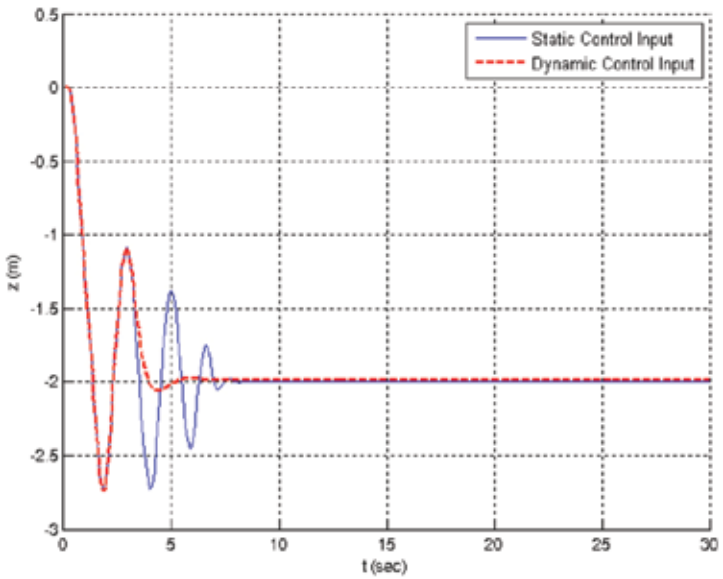


Fig. 8. Online Overshoot &Undershoot Suppression after 4 seconds

4.2 Simulation condition 2

Similar to the previous simulation, PC1 running the controller gain and system matrix with assuming all terms in $Q \rightarrow 0$ and $R \rightarrow 0$, except: q_{11} = 205.21, q_{44} = 131.31, and r_{11} = 400, as illustrated in Figure 9 using solid line. To reduce the rise time duration in the runtime simulation, when $t = 2$ seconds PC2 send a new weighting matrices to the PC1 to change a pitch angle deviates to 10° from zero with 0.02 meter deviation at -5 meter depth: q_{11} = 205.21, q_{44}= 23.83, and r_{11} = 2500, as illustrated in figure 9 using dash line. In this case, a settling time response also improved easily during runtime when $t \geq 2$ seconds. It could be seen that time response is improve significantly.

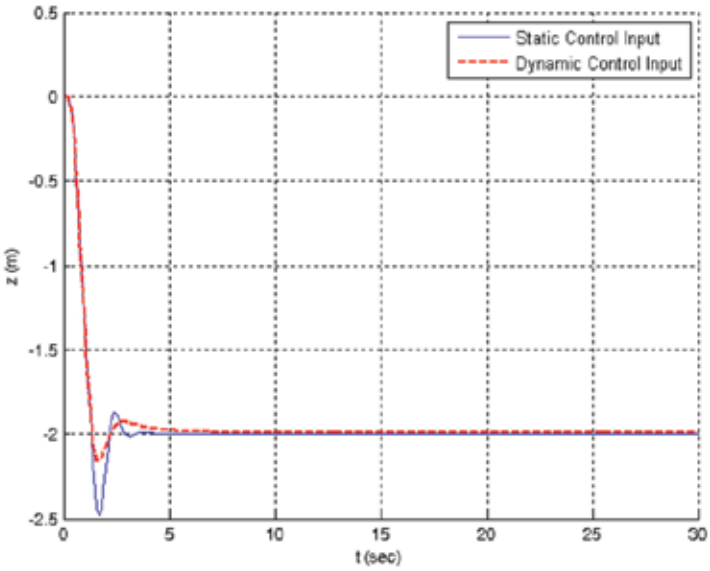


Fig. 9. Online Settling Time Suppression after 2 second.

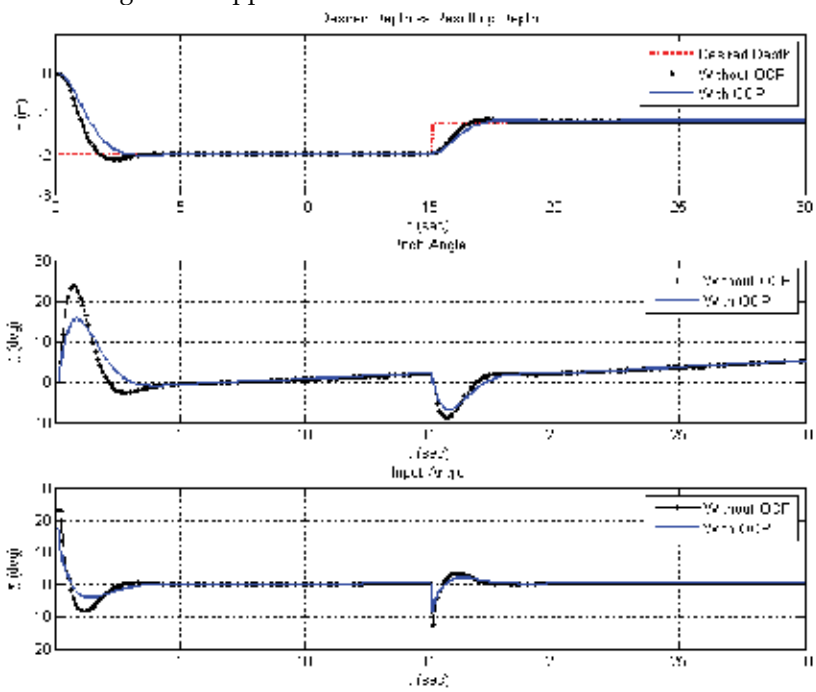


Fig. 10. Depth Control Simulation Result

4.3 Simulation condition 3

In the first simulation, we perform a static simulation without OCP infrastructure using single weighting matrices, we assume all terms in $Q \rightarrow 0$ and $R \rightarrow 0$, except: q_{11} = 205.21, q_{44}= 23.83, and r_{11} = 2500, as illustrated in figure 10 using dot line. In the second simulation, PC2 send any weighting matrices value to the PC1 (i.e. Q=diag[2,10,20,90], and R=[0.1]). Then GA will

find a new weighting matrices, a global optimal value in this runtime all terms in $Q \rightarrow 0$ and $R \rightarrow 0$, except: q_{11} =397.33, q_{44} =98.67 and r_{11} = 1272.8, as illustrated in figure 10 using solid line.
We have 2 simulation results, with OCP and without OCP, but the performance is similar as depicted in figure 10 because of given system is simple. If the system is more complex using 6DOF with more sensors and actuators we can see the difference performance that OCP is more effective in complicated cases.

5. Concluding remarks

In this paper, we propose a new approach of decentralized system environment for AUV simulation using Matlab and CORBA event channel coexistence on several machines, we believe it will emerge more investigation how the real-time control system performance could be reconfigured easily both in semi-automatically or manually interventions by a remote station.
In the future research, especially we expect to uncover the effective CORBA programming to support Matlab and CORBA event channel coexistence that will be affected to increase the degree of real-time reconfigurable control significantly.

6. References

Cristi, R., Papoulias, F.A., and Healey, A.J. (1990), Adaptive Sliding Mode Control of Autonomous Underwater Vehicles in the Dive Plane, *IEEE Journal of Oceanic Engineering*, Vol. 15, Issue: 3, page(s): 152-160, Piscataway, NJ, USA.

Healey A.J., Papoulias P.A., and Cristi R. (1989), Design and Experimental Verification of a Model Based Compensator for Rapid AUV Depth Control, *Proceeding of the Int'l Symposium on Unmanned Untethered Submersible Technology*, page(s): 458-474, Washington DC, USA.

Paunicka J.L., Corman W.E.,and Mendel B.R. (2001), A CORBA-Based Middleware Solution for UAVs, *Proceeding of Fourth International Symposium on Object-Oriented Real-Time Distributed Computing*, page(s): 261-267, Magdeburg, Germany.

Riedel, J.S., (1993), Pitchfork Bifurcations and Dive Plane Reversal of Submarines at Low Speeds, *Engineer's Thesis*, Naval Postgraduate School, Monterey, California, USA.

Schmidt D. C., and Kuhns K. (2000), An Overview of the Real-time CORBA Specification, *IEEE Computer special issue on Object-Oriented Real-time Distributed Computing*. Vol. 33, Issue: 6, page(s): 56-63, Los Alamitos, CA, USA.

Sname, The Society of Naval Architects and Marine Engineers (1950), Nomenclature for Treating the Motion of a Submerged Body Through a Fluid. *Technical Research Bulletin*, No. 1-5, NY, USA.

Valavanis K.P., Gracanin D., Matijasevic M., Kolluru R., and Demetriou G.A. (1997), Control architectures for autonomous underwater vehicles, *IEEE Control Systems Magazine*, Vol.17, Issue: 6, page(s): 48-64. Ann Arbor, MI, USA.

Wills L., Kannan S.K., Heck B.S.,Vachtsevanos G., Restrepo C., Sander S., Schrage D.P., and Prasad J.V.R. (2000), An Open Software Infrastructure for Reconfigurable Control Systems, *Proceeding of American Control Conference,* Vol. 4, page(s): 2799-2803, Chicago, IL, USA.

Wills L., Kannan S.K., Sander S., Guler M., Heck B.S., Prasad J.V.R., Schrage D.P., and Vachtsevanos G. (2003), A Prototype Open-Control Platform For Reconfigurable Control Systems, In: *Software-Enabled Control: Information Technology for Dynamical Systems*, Samad T. and Balas G. ,page(s):63-84, John Wiley & Sons/IEEE Press, ISBN 0-471-23436-2, Hoboken, NJ, USA.

26

Autonomous Underwater Gliders

Wood, Stephen
Florida Institute of Technology
United States of America

1. Introduction

Over the past few decades, a range of strategies and techniques has been used to monitor the sea. More recently, the role of monitoring has been expanded to include the use of autonomous underwater vehicles to perform ocean surveys. With these vehicles it is now possible for the scientist to make complex studies on topics such as the effect of metals, pesticides and nutrients on fish abundance, reproductive success and ability to feed, or on contaminants such as chemicals or biological toxins that are transported in particulate form and become incorporated into living organisms (plankton, bivalves, fishes) or become deposited in bottom sediments. The scientist or environmentalist may desire to detect hazardous substances in the ocean such as chemicals from an underwater vent or toxic algae such as red tide. Additionally, the military's detection of mines, biological, chemical or radioactive threats are also very important in the monitoring of the seas.

These considerations explain today's development of new types of autonomous underwater vehicles with integrated sampling equipment that is able to perform a wide-range of fully automated monitoring surveys over extended periods of time. These vehicles survey and monitor the sea environment in a cost-effective manner combining survey capabilities, simultaneous water sampling and environmental data gathering capacities. Included in these types are autonomous underwater gliders that have the ability to glide for long distances and are in some cases able to travel under power. There are currently four classes of underwater gliders: 1) those that use mechanical or electrical means of changing their buoyancy (i.e., drop weights, or electrical power from batteries), 2) those that use the thermal gradient of the ocean to harness the energy to change the vehicle's buoyancy, 3) those that are able to use other means of power such as ocean wave energy, and 4) hybrid vehicles that use standard propulsion systems and glider systems.

Gliders are designed for deep water where the vehicle can traverse large areas with minimal use of energy and are specifically designed for the needs of the Blue Water scientist, which require greater control over the vehicle (the free-drifting profiling Argos floats that scientists often use have no control capabilities beyond descending and rising vertically in the water column. The newest Argos float model cycles to 2000m depth every 10 days, with 4-5 year lifetimes for individual instruments)(Argos, 2008). Some of these glider AUVs have space for multiple scientific instruments and have the ability to obtain water or biological samples. Scientists who perform experiments in shallower water can also use the vehicle for short duration gliding dives or under power if one of the hybrid gliders is used.

The more information scientists are able to accumulate the better they will be able to determine the health of the ocean ecosystem and document the specific ecosystem parameters. Using an AUV glider, pollution of ocean waters can be detected and quantified in an automated way; depending on the glider, water samples can be taken and analyzed to determine water quality as well as any contaminating chemicals. Thus, dangerous substances in the sea can be detected earlier and their harmful effects can be dealt with quicker. Depending on the vehicle's configuration the scientist may have the ability to take fly-by photographs of organisms in the water column.

Initially gliders were targeted for missions that were a combination of three archetypes: time series, transects and roving assistants to research cruises (Sherman et al., 2001) by the scientist for surveying and monitoring the deep-sea environment. This is still true but scientists at various institutions such as Florida Institute of Technology's Department of Marine and Environmental Systems in Melbourne Florida have desired more. A survey was conducted of the opinions of marine scientists (biological, physical, chemical oceanography, marine biology, environmental science, and ocean engineers) and the biological research published on the Internet with respect to which organisms take precedence in ocean studies was analyzed. From these investigations, one of the most important biological groups in the life cycle of higher ocean organisms (e.g., fish), and consequently a very important element in the research of all marine organisms, was found to be the phytoplankton[1]. Phytoplankton play a fundamental role in the ocean's biological productivity and directly impact the climate. It is important that scientists determine how much phytoplankton the oceans contain, where they are located, how their distribution is changing with time, how much photosynthesis they perform, and what organisms such as marine invertebrate larvae feed upon the phytoplankton (Herring, 2007b).

Next, the marine invertebrate larvae and zooplankton (e.g. krill - *Euphausia superba*) were found to be a very important biological group that affect the life cycle of higher ocean organisms. Krill are small shrimp like crustaceans that are the most important zooplankton species associated with sea ice and are very important in the Antarctic food web. Krill occur in groups or large swarms and occupy a niche similar to that of the herring in the North Atlantic. Krill attain a size of 6-cm and feed primarily on phytoplankton or sea ice algae. Its feeding apparatus is built to filter phytoplankton out of the water column and to scrape algae from the ice. Krill is the staple food of many fish, birds and mammals in the Southern Ocean. The biomass of Antarctic krill is considered to be larger than that of the earth's human population and krill swarms can occupy an area equivalent to 2.5 times the size of Washington, DC (AWI, 2008a).

Southern Ocean GLOBEC is conducting a study of the Antarctic krill in which they are attempting to define the habitat, prey, predators, and competitors of this species. This organization could make immediate use of such a vehicle as the autonomous underwater glider. In fact, recent evidence indicates that seasonal coverage is necessary to fully understand the linkages between the environment, krill, and top predators.

---

[1] Phytoplankton are microscopic plants that live in the ocean. There are many species of phytoplankton that grow abundantly in oceans around the world and are the foundation of the marine food chain. Since phytoplankton depend upon certain conditions for growth, they are a good indicator of change in their environment making them of primary interest to oceanographers and environmental scientists (Herring, 2007a).

The zooplankton science questions that a glider could help answer are (AWI, 2007b):

1. What is the abundance of krill?
2. How many populations are there?
3. How do krill survive during winter with a minimal food supply?

A third important biological group that affects the life cycle of higher ocean organisms is the algae, which at times is responsible for harmful algal blooms (HAB). HABs occur throughout the world, affecting European and Asian fisheries, Caribbean and South Pacific reef fishes, and shell fishing along the coasts of the United States. These HABs are caused by several species of marine phytoplankton, microscopic plant like cells that produce potent chemical toxins (Mote, 2007).

Research on these and other biological groups requires non-traditional approaches to acquire the needed scientific information. Various institutions are addressing this issue by developing vehicles which implement special biological catching and photographing systems to document small visible species, using a navigation system that will use the scientific data to control the vehicle's movement.

In addition to biological investigations, documenting the chemical make-up of all areas surveyed, specifically where samples were taken is important for the scientist to obtain a complete understanding of that region. The chemical layout and the corresponding biological data are normally for a specific transect, but a vehicle's transect might not be on a traditional grid pattern. For example, the scientist may desire to obtain samples within a polluted area with a specific concentration of the pollution. To accomplish this, non-traditional approaches of navigation are required to acquire the desired scientific information. Data from geophysical and acoustic sensors can be combined, analyzed and entered into the navigation system to aid in controlling the vehicle with respect to the chemical information supplied.

To date, most survey AUVs have relied on rudimentary single variable differential gradient navigation systems, external triangulation, or inertial based dead reckoning systems. Research is now being conducted using the changes in various geophysical parameters as navigation cues (i.e., phenomenon based navigation). Some of these parameters are temperature, salinity, turbidity, chlorophyll, rhodamine, fluorescein, and passive acoustic signals. These navigation techniques are expected to provide a better understanding of the geophysical environment where biological samples are obtained, in addition to characterizing the data.

2. Vehicles

Looking at the current situation in AUV technology (see Figure 1), there are currently over 50 different types of AUVs in research and commercial operation, just a few of these systems are: Hydroid's REMUS (USA), Bluefin Robotics Corporation's Odyssey (USA), Woods Hole's ABE (USA), FAU's (USA), Boeing's, Oceaneering's and Fugro's Echo Ranger (USA), Kongsberg Simrad's Hugin 3000 (Norway), Sias-Patterson's Fetch (USA), University of Southampton's AUTOSUB (England), Alive and Swimmer (France), and Hafmynd's Gavia (Iceland). Of these autonomous underwater vehicles all but a few of them are 100 percent powered. Three of these vehicles are torpedo shaped (see Figure 2) and move without power. These are the Webb Research Corporation's Slocum glider (the name Slocum commemorated the first person who sailed around the world solo, Joshua Slocum), University of Washington's Applied Research Laboratory's "Seaglider", and Scripps Institution of Oceanography "Spray" (Spray was the name of Joshua Slocum's boat when he sailed around the world) currently sold by Bluefin

Robotics Corporation. These three vehicles are from the United States of America and have the ability to do studies in glide mode.
These vehicles glide slowly down to a specified depth and then back to the surface using a buoyancy control system tracing a saw-tooth profile, observing data such as temperature and conductivity versus depth. When the vehicle is at the surface, positioning is obtained via GPS and communication between the vehicle and the home base is via satellite. The three gliders are small semi-torpedo shaped AUVs that control their forward motion by the glide path taken and changing buoyancy. Wings allow steerable gliding, thus horizontal propulsion.

Fig. 1. 1) Woods Hole Oceanographic Institution, USA: Autonomous Benthic Explorer (ABE), 2) ECA, France: ALISTAR 3000, 3) Boeing, Oceaneering and Fugro, USA: ECHO RANGER, 4) Bluefin Robotics, USA: BLUEFIN 21, 5) Southampton Oceanography Center, Great Britain: AUTOSUB, 6) Florida Institute of Technology, USA: TUVAAQ, 7) Kongsberg-Simrad, Norway: HUGIN 3000, 8) Atlas Maridan, Germany: M600, 9) Hydroid, USA: REMUS, 10) Cybernetix, France: ALIVE, 11) Cybernetix, France: SWIMMER, 12) Autonomous Undersea Systems Institute, USA: SAUV.

The Slocum glider, the Seaglider and Bluefin's glider Spray are excellent at the tasks they are designed to do (i.e., very long term, very little power, slow cruising of the ocean's water column), but they are limited as to the type of payloads they can carry, and they have no active propulsion for times that require more than buoyancy thrust (e.g., control of the vehicle at the surface). These autonomous underwater gliders each change their buoyancy to be able to travel horizontally in the ocean's water column using the lift on their wings, like a

normal glider does to convert vertical velocity into forward motion. These vehicles are not capable of traveling in a horizontal path as would a typical propeller vehicle but follow a saw-tooth path as the vehicle descends or ascends.

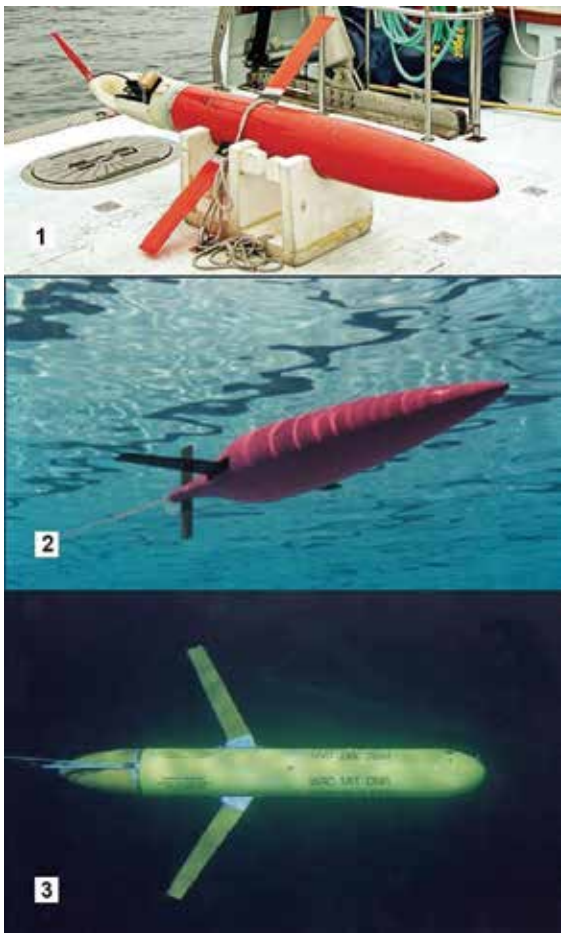


Fig. 2. 1) Spray, 2) Seaglider, 3) Slocum Glider (Photos courtesy of Bluefin, Applied Physics Laboratory, University of Washington and Webb Research Corporation)

These three underwater gliders were designed specifically for long term sampling and easy deployment and recovery by a minimal crew (i.e., one to three people) on any size boat or ship. Consequently, this requires a design that has minimal space for instrumentation and is limited in its function/capabilities. These vehicles are relatively inexpensive, typically less than the cheapest powered AUV's (e.g., $100,000 for the EcoMapper AUV by YSI Inc.), costing less than a week and a half of ship time for a research vessel.

With respect to these gliders four basic sampling modes exist: 1) vertical sampling where the forward motion of the vehicle counters any local currents to maintain position, 2) horizontal saw-tooth sampling where the forward motion allows for the vehicle to obtain information both vertically and horizontally, 3) array sampling where multiple gliders form a distribution of sampling instruments covering an entire region, and 4) long life and repeat sampling over an extended duration.

Spray and Slocum Battery/Electric Gliders

The Slocum Battery (Webb *et al*., 2001) and Spray Gliders (Sherman *et al*., 2001) have been optimized for missions in shallow coastal environments. Each of these vehicles uses battery power to control the buoyancy.

The Slocum Battery is controlled by different methods. The pitch and roll is controlled by translating and rotating the internal battery packs. A rudder controls the turning rate and the pitch moment and the buoyancy at the surface are aided by the inflation of an airbladder. The Slocum battery uses an efficient shallow water single stroke pump to move water in and out of the vehicle for volume control. The communication and GPS antennas are embedded in a vertical stabilizer, which rises above the ocean surface when the vehicle is pitched forward (Griffiths, 2002). Additionally, "the yaw moment for steering is achieved by mounting the wings aft of the center of buoyancy, and when rolled, the lateral component of lift creates a yaw moment" (Webb *et al*., 2001).

Slocum Battery/Electric Specifications (Webbresearch, 2008a)(Griffiths, 2002)

- Weight: 52 kg
- Hull Diameter: 21.3 cm
- Vehicle Length: 1.5 m
- Wing Span: 120 cm
- Depth Range: 4 – 200 m (coastal model or 1000 m (1 km model)
- Payload: 3 to 4 kg
- Speed: 0.3 to 0.4 m / sec
- Energy: alkaline batteries
- Range: 1500 km
- Navigation: GPS, internal dead reckoning, altimeter
- Sensor Package: conductivity, temperature, depth
- Communications: RF modem, Iridium satellite, ARGOS, Telesonar modem

Spray (Spray, 2008a)

- Weight: 51 kg
- Hull Diameter: 20 cm
- Vehicle Length: 2 m
- Wing Span: 110 cm
- Depth Range: 200 – 1500 m
- Payload: 3.5 to 51.8 kg depending on the glide ratio between 19 and 25 degrees
- Speed: 0.25 – 0.35 m/sec horizontal
- Energy: Primary lithium sulfuryl chloride batteries
- Range: 3500 to 4700 km depending on the glide ratio.
- Navigation: GPS, and internal dead reckoning, altimeter
- Sensor Package: Sensors used include: Precision Measurement Engineering CTD; modified Sea Bird 41 CP CTD with seawater pump; Sea Point Optical Backscatter Sensor; Sea Point Chlorophyll Fluorometer; Tritech PA200 acoustic altimeter for bottom avoidance; and Sontek Argonaut-SGP 750-kHz Acoustic Doppler Current Profiler.
- Communication: Iridium satellite

The Spray glider (see Figures 3 and 4) is similar to the Slocum Battery Glider but with a more hydrodynamic shape giving it about fifty percent less drag than the Slocum Battery (Sherman et al., 2001). The Spray was targeted for long-range up to 4700 km and down to 1500 meters depth by optimizing the use of energy. Use of a high-pressure reciprocating pump with external bladders makes the vehicle similar to the ALACE floats (Davis et al., 1992).

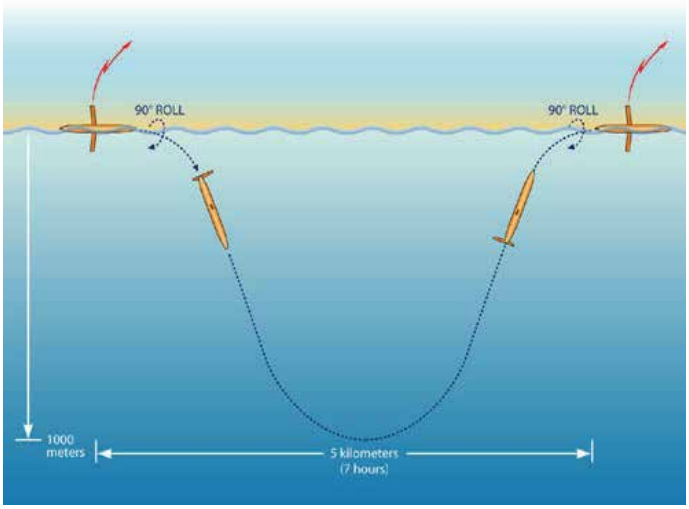


Fig. 3. Spray's method of travel (Spray, 2008b)

The "glide control in Spray is achieved exclusively by axial translation and rotation of internal battery packs. Pitch is controlled simply by moving the center of gravity in the manner of a hang glider. Turning is initiated by rolling. This gives the lift vector a horizontal component and induces vehicle sideslip in the plane of the wing in the direction of the buoyant force. The horizontal component of lift provides the centripetal force for turning while sideslip acting on the vertical stabilizer produces the yaw moment needed to change vehicle heading. For example, to turn right during descent the right wing is dropped, like a conventional airplane, generating a lift component to the right that drives the vehicle to the right. Sideslips down and to the right acts on the vertical stabilizer causing the nose to yaw to the right. To turn right in ascent the glider is rolled oppositely by dropping the left wing" (Griffiths, 2002).

Additionally, the vehicle must rotate 90 degrees to present the GPS and satellite communication antennas housed in a wing. Payload is mounted either in the expandable aft flooded component section or on the hull.

Seaglider & Deepglider

The Seaglider and Deepglider (see Figures 5, 6 and 7), commercially sold by iRobot (Bedford MA, USA), are similar to the Spray and Slocum Battery gliders. The Seaglider and Deepglider are identical in looks but the Deepglider is made out of a composite pressure hull of thermoset resin and carbon fiber making it capable of diving to a depth of 6000 meters (Osse & Lee, 2007)(Osse et al., 2007). The Seaglider using an efficient use of energy allows it to operate one-year 4600 km missions (Eriksen *et al*., 2001). Seaglider uses a hydrodynamic aluminum pressure hull that is contained within a free-flooded fiberglass

fairing[2] that supports the wings. The flooded aft section is used to carry self-contained instruments on both the Seaglider and Deepglider. Both vehicles have a trailing antenna rod and the fairing encloses the pressure hull. In weak currents the vehicles can maintain position by pitching vertically with minimal buoyancy. As with the Slocum series and Spray gliders, the Seaglider and Deepglider control their buoyancy with a hydraulic system similar to the ALACE system. In both the Seaglider and Deepglider it is the movements of internal masses (i.e., batteries) which control the pitch and yaw of the vehicle while gliding, and also raise the antenna for communication and GPS navigation. Another interesting aspect of these two vehicles is that due to the lifting wings being so far aft the turning method is opposite to what one would expect (i.e., opposite to the Slocum and Spray gliders). To turn right while descending the left wing is lowered so that the wing lift pushes the stern left, "overcoming lift off the vertical stabilizer, and initiating a turn to the right. Hydrodynamic lift on the side slipping hull produces the centripetal force to curve the course. Conversely, in ascent a roll to the left produces a left turn" (Griffiths, 2002).
The Seaglider has made thousands of dives since its inception in 1999. Some of these dives can be seen on Seaglider's website: http://www.apl.washington.edu/projects/seaglider/summary.html. The first Deepglider tests were made in November 2006 off the Washington state coast where it made test dives for 39 days with dives down to 2713 meters depth and a lateral distance of 220 km.

Seaglider (Griffiths, 2002)(APL, 2008) (Osse et al., 2007)

- Weight: 52 kg
- Hull Diameter: 30 cm
- Vehicle Length: 1.8 m
- Wing Span: 1 m
- Depth Range: 1000 m
- Payload: 25 kg
- Speed, projected: 0.25 m/sec (1/2 knot) horizontal
- Energy: Lithium primary batteries
- Energy: Lithium primary batteries
- Range: 4600 km (3800 km proven mission)
- Navigation: GPS, and internal dead reckoning, altimeter
- Sensor Package: Sea-Bird temperature-conductivity-dissolved oxygen, Wet Labs fluorometer-optical backscatter
- Communication: Iridium satellite

Deepglider (Osse et al., 2007)

- Weight: 62 kg
- Hull Diameter: 30 cm
- Vehicle Length: 1.8 m
- Wing Span: 1 m
- Depth Range: 6000 m
- Payload: 25 kg
- Speed, projected: 0.25 m/sec (1/2 knot) horizontal
- Energy: Lithium sulfuryl chloride batteries

[2] A **fairing** is a structure whose primary function is to produce a smooth outline and reduce drag

- Range: 8500 km
- Navigation: GPS, and internal dead reckoning, altimeter
- Sensor Package: Sea-Bird temperature-conductivity-dissolved oxygen, Wet Labs fluorometer-optical backscatter
- Communication: Iridium satellite

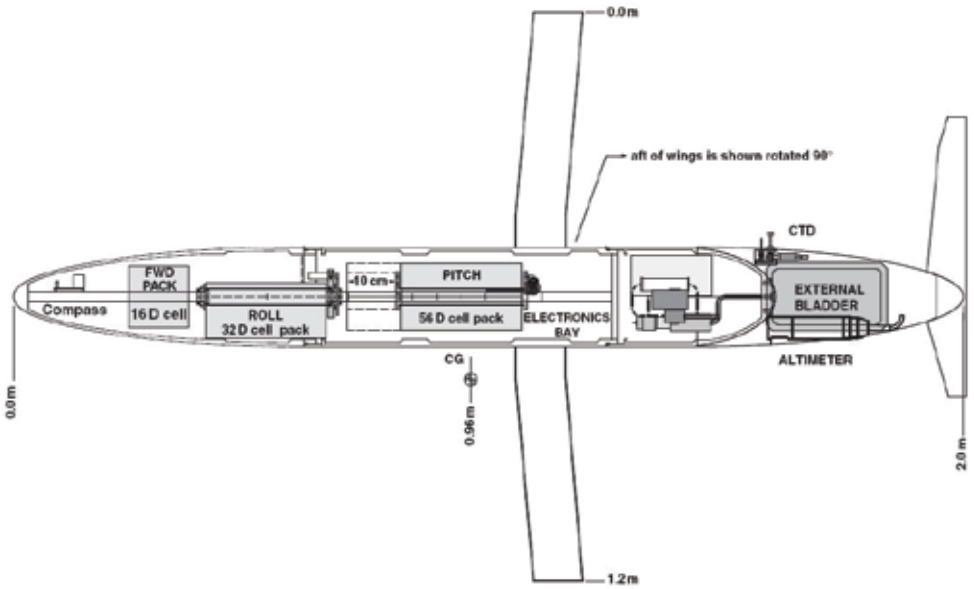


Fig. 4. Spray Schematics (Spray, 2008b)

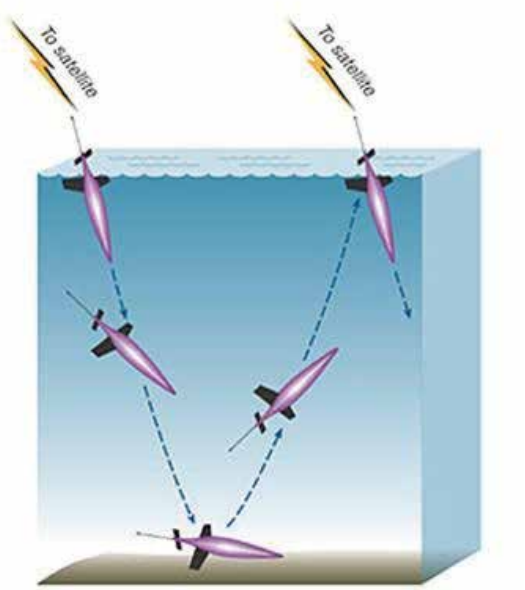


Fig. 5. Seaglider's method of travel (Seaglider, 2008)

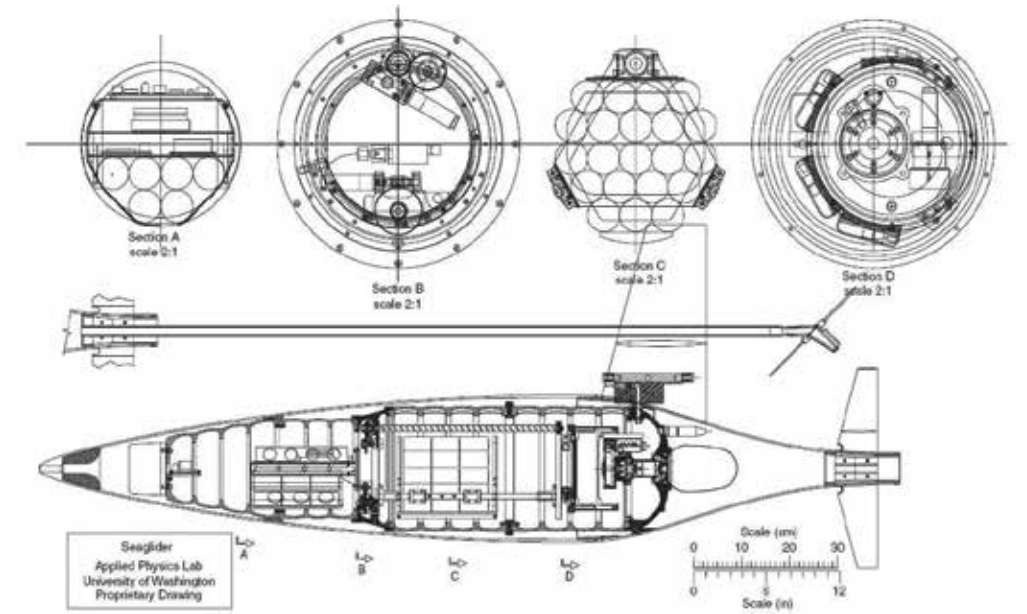


Fig. 6. Seaglider Schematic (Griffiths, 2002)

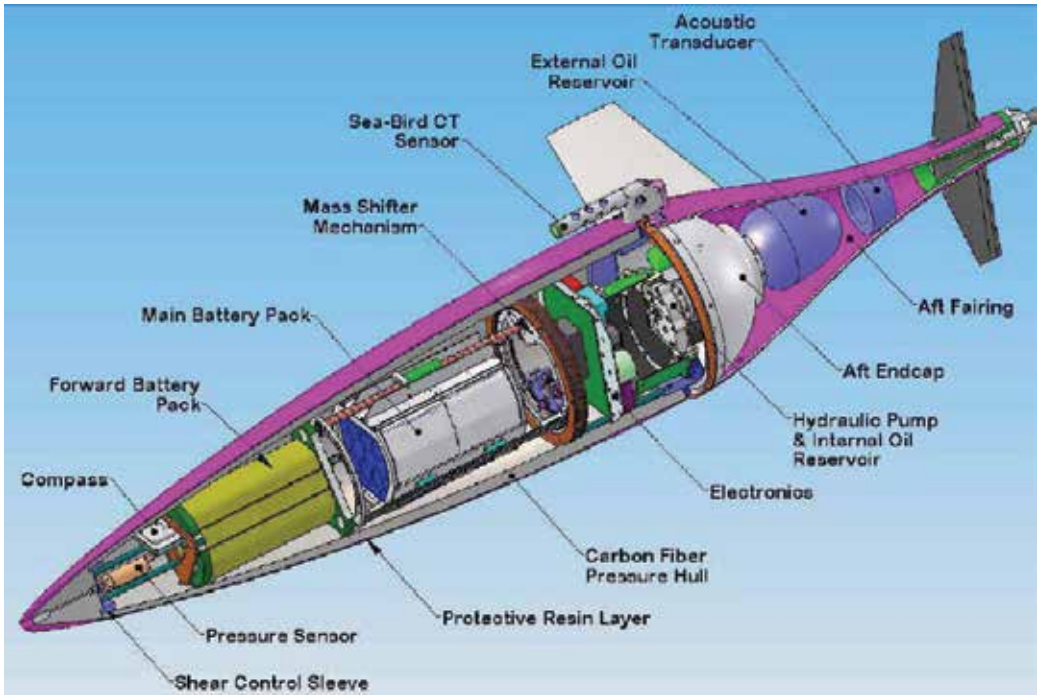


Fig. 7. Deepglider (Osse et al., 2007)

Slocum Thermal Glider

The Slocum Thermal glider (see Figures 8 and 9) was developed and optimized for long duration missions with a well-developed thermocline. The propulsion of the vehicle is

derived from harnessing the energy of the thermal gradient between the ocean's surface and bottom for use as the vehicle's propulsion. "In missions with electric-powered gliders, 60--85% of the energy consumed goes into propulsion, so a thermal-powered glider may have a range 3--4 times that of a similar electric-powered vehicle. Except for its thermal buoyancy system and using roll rather than a movable rudder to control turning, Slocum Thermal is nearly identical to Slocum Battery" (Griffiths, 2002).

The Slocum Thermal glider uses the change in volume from a material's (ethylene glycol) freezing and melting as the means of vehicle propulsion. The vehicle begins to descend by venting the external bladder into an internal bladder using the pressure difference between the two chambers (i.e., the hull/internal bladder, filled with Nitrogen, is slightly below atmospheric pressure). As the vehicle passes through the freezing point of the material during its descent the contraction of the material causes the fluid in the internal reservoir to be drawn out into a heat exchanger. To ascend the pressurized material in the heat exchanger is transferred to the external bladder causing the vehicle to switch from negative to positive buoyancy. As the vehicle ascends the warming of the ocean waters cause the material to melt and expand further increasing its buoyancy. The vehicle arrives at the surface with the same conditions it had at the start, i.e. in a stable thermal equilibrium with the external bladder inflated, the material expanded, and the internal bladder at a slightly negative pressure. The material and pressurized nitrogen is at a slightly greater pressure than the external ocean pressure. The thermodynamic stages of the system can be seen in Figure 10.

Slocum Thermal (Webbresearch, 2008b)

- Weight: 60 kg
- Hull Diameter: 21.3 cm
- Vehicle Length: 1.5 m
- Wing Span: 120 cm
- Depth Range: 4 – 2000 m
- Payload: 2 kg
- Speed: 0.4 m/sec horizontal (projected)
- Energy: Thermal engine, Alkaline batteries for instruments, communication and navigation
- Endurance: 5 years
- Range: 40,000 km
- Navigation: GPS, internal dead reckoning, altimeter
- Sensor Package: conductivity, temperature, depth
- Communications: RF modem, Iridium satellite, ARGOS

The Spray, Slocum (Battery & Thermal), Seaglider and Deepglider are very similar in size and general characteristics. They were designed with the same objectives, specifically in being small and easily deployed and recovered by only a couple of people. The vehicles were to be slow and the propulsion using only buoyancy control envisioned by Douglas Webb and Henry Stommel. The vehicles are dependent on the energy efficiency and glide trajectory angle during each traverse to monitor the ocean. Currently, various institutions (e.g., the University of Southampton, Great Britain) are starting the investigation of long-duration, highly efficient, slow-speed, powered autonomous underwater vehicles. These investigations will lead to the development of new highly optimized efficient wings. The optimum vehicle to handle a saw-tooth method of data sampling, as well as a vertical and

horizontal means of sampling will be some form of hybrid vehicle with a glide and a power mode that takes each sampling means into account.

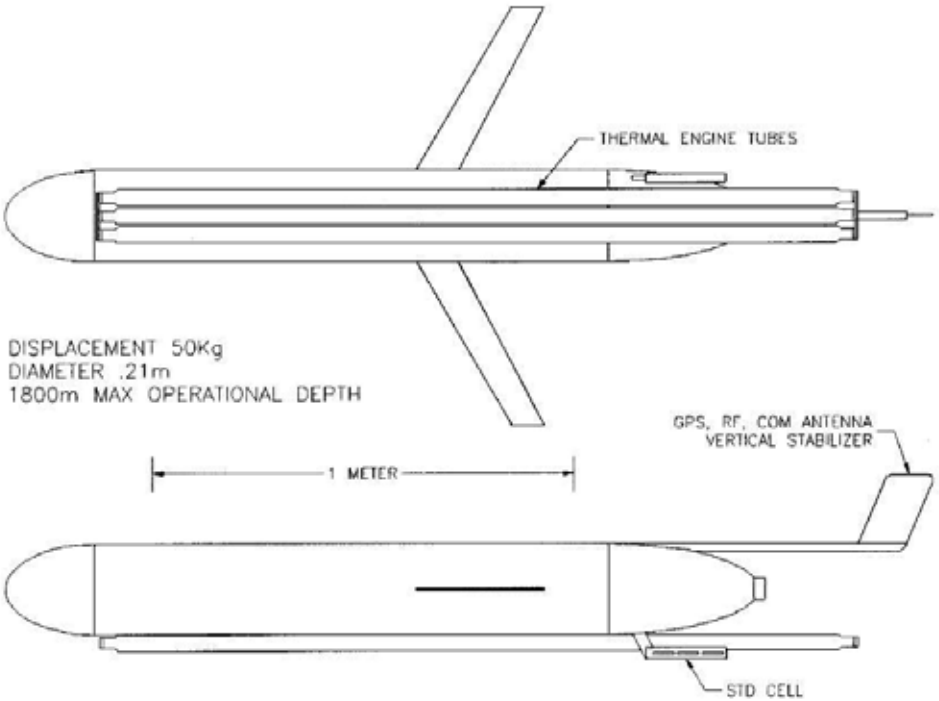


Fig. 8. Slocum Glider Schematic (Webb et al., 2001)

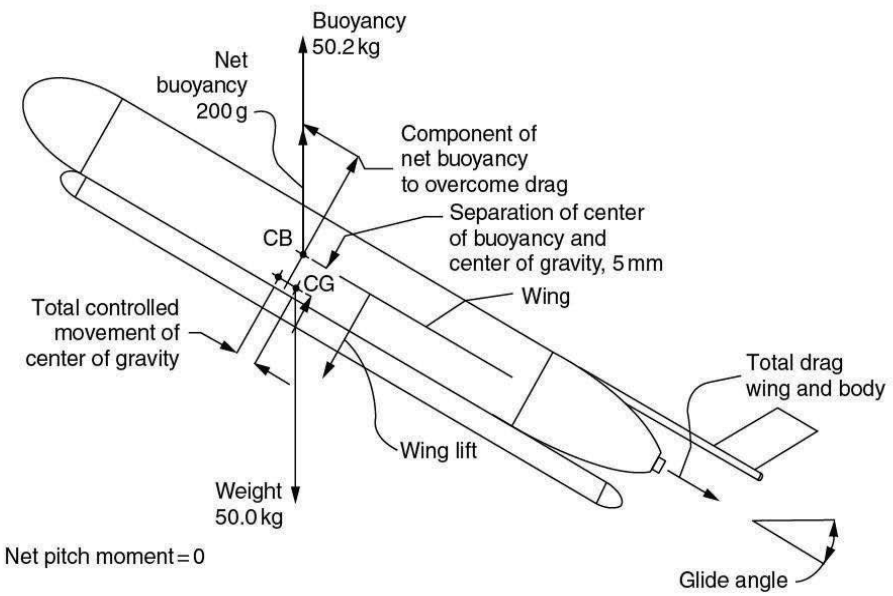


Fig. 9. Slocum Thermal - Gliding forces on the vehicle (Webb et al., 2001)

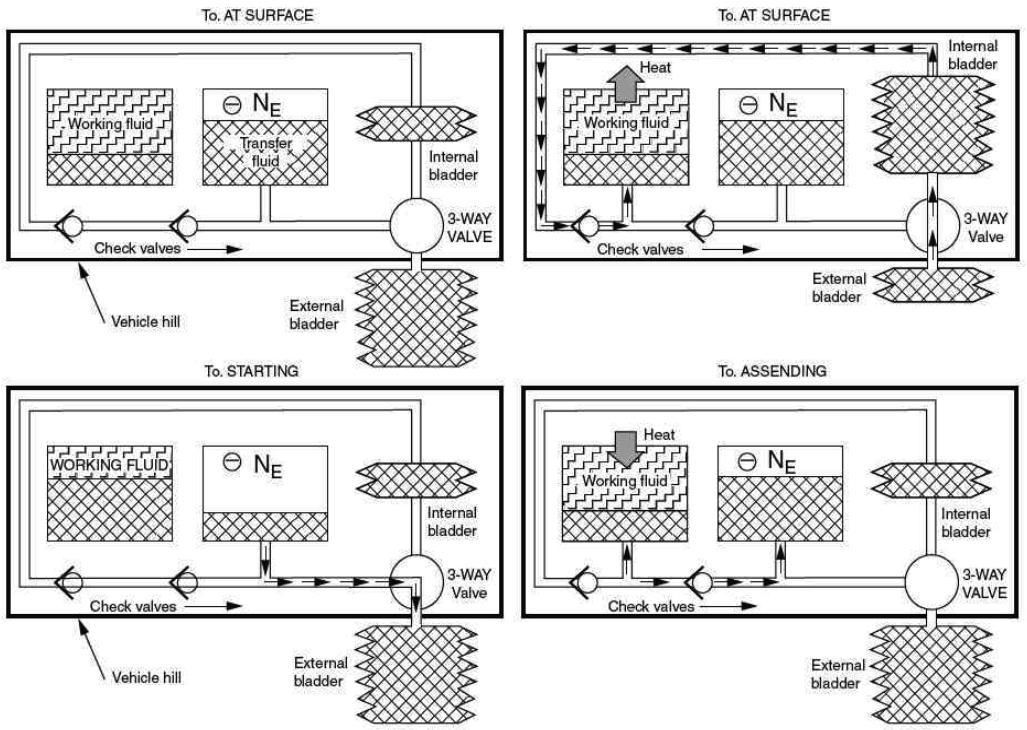


Fig. 10. Slocum Thermal Cycle (Webb et al, 2001)

3. Military vehicles

The military has developed an advanced underwater winged glider based on the air force's Flying Wing design, the Liberdade XRAY (see Figure 11). This vehicle is "being developed as a part of the Navy's Persistent Littoral Undersea Surveillance Network (PLUSNet) system of semi-autonomous controlled mobile assets. PLUSNet uses unmanned underwater vehicles (UUVs) and autonomous underwater vehicles (AUVs) to monitor shallow-water environments from fixed positions on the ocean floor or by moving through the water to scan large areas for extended periods of time" (ONR, 2006).

The XRAY was develop primarily with the aid of the Marine Physical Laboratory at Scripps Institution of Oceanography and the University of Washington's Applied Physics Laboratory, and also with the following institutes, universities and corporations: University of Texas at Austin's Applied Research Lab, Applied Research Lab at Penn State University, MIT, Woods Hole Oceanographic Institute, Harvard University, SAIC, Bluefin Robotics, Metron, Heat, Light, and Sound (HLS) Research, and the Space and Naval Warfare (SPAWAR) Systems Center in San Diego.

The vehicle is the largest of all of the underwater gliders (6.1 meter wing span), which is an advantage in terms of hydrodynamic efficiency and space for energy storage and payload. The glider's primary function is to track quiet diesel–electric and the new fuel cell submarines operating in shallow-water. According to military doctrine it can "be deployed quickly and covertly, then stay in operation for a matter of months. It can be programmed to monitor large areas of the ocean (maximum ranges exceeding 1000 km with on-board

energy supplies). The glider is very quiet, making it hard to detect using passive acoustic sensing" (ONR, 2006).

Fig. 11. XRAY Glider (APL, 2007)

The vehicle was designed for easy and rapid deployment and retrieval, as well as payload carrying capability, cross-country speed, and horizontal point-to-point transport efficiency which is better than existing gliders. Liberdade XRay's first major ocean test was performed in August 2006 in Monterey Bay, California, where it reported real-time via an 3.0 to 8.5 kHz underwater acoustic modem as well as with an Iridium satellite system while on the surface. The vehicle had an array of 10 kHz bandwidth hydrophones located in the SONAR dome and across the leading edge of the wing. The XRay exceeded a 10 to 1 glide slope ratio (D'Spain et al., 2007). Later deployments were in the Philippine Sea, near Hawaii, and in Monterey Bay using the hydrophone array "to detect low frequency source signals, marine mammals (blue and humpback whales), and ambient ocean noise" (APL, 2007). The XRay glider is hoped to achieve 1–3 knot cruise speeds, have a 1200–1500 km range, and be able to remain on-station up to 6 months in partial buoyant glides.

4. Other vehicles

WaveGlider

Another vehicle that will soon come to market is Liquid Robotics' entirely new autonomous ocean vehicle "WaveGlider" that harvests all of its energy from waves and sun. The concept is a shallow water vehicle that uses the ocean waves as its primary energy source to propel it through the water. During the spring and summer of 2008 the WaveGlider underwent extended periods of field testing in the Pacific Ocean.

The design consists of a surface float (similar to a surfboard) that is tethered to a sub-surface glider about 7 meters below the surface. This subsurface glider looks similar to the Slocum glider (i.e., a torpedo hull with a simple rudder), except instead of one pair of wings there

are six sets of wings down the vehicle's side. The wings have a mechanism that "ratchet" in such a way that when a wave at the surface lifts the float, the entire system (float and glider) rises while the wings stay horizontal. As the wave passes by, the glider sinks and the wings pivot to create a downward pitch which causes the glider to fly forward and slide downward at an angle. Because the float and glider are tethered together the glider will stop at the end of the line's reach causing the surface float to move forward. Consequently, the whole system moves forward in a "saw-tooth" pattern corresponding to the waves. The surface-float shoots forward in small bursts across the water controlled by the rudder. The vehicle requires at least seven (7) meters of water and a minimum wave height to operate. It has high-endurance, is able to station-keep and the method of movement allows it to move in any direction regardless of wave direction. The vehicle does not "surf" the wave, consequently, it can traverse up a wave. All it needs is the up and down motion that translates into forward motion of the vehicle. The vehicle moves quite slowly[3] and high currents are a problem.

The WaveGlider's surface float houses most of the electronics (i.e., navigation and communication equipment) along with solar cells to recharge the electronic battery packs. Only wave motion is used for propulsion. The vehicle is quite remarkable and Harbor Branch Oceanographic Institute is expected to develop a mobile observatory, in other words, a distributed sensor network for surface sensing using these vehicles. Additionally, they are hoping to demonstrate the swarming technologies that the engineering division at Harbor Branch has been working on with these vehicles (Frey, 2008).

ALBAC

One of the first gliders, the ALBAC conducted sea trials at the Suruga Bay of Japan in 1992. The vehicle, developed at the University of Tokyo in the lab of Tamaki Ura, does not have an active buoyancy control system, but a simple drop weight system with only one glide cycle.

> The "ALBAC has fixed wings and a vertical and horizontal tail. It is 1.4 m long," 120 cm in wide, "weighs 45 kg, and can dive to depths of 300 m at speeds of one to two knots (.5 to 1.0 m/s). It has horizontal tail fins which change angle at inflection from downwards to upwards gliding, a feature not present in other gliders. The wings and tail are larger in comparison to the body than on Slocum, Spray or Seaglider. ALBAC moves a battery pack internally to control pitch and yaw in the same manner as Seaglider. Because it has no ballast pump, ALBAC carries batteries to power only its instruments and actuators.
>
> ALBAC carries flight sensors including compass, depth, pitch, roll, and a propeller-type velocity meter. Note, that Slocum, Spray and Seaglider do not carry velocity meters in order to conserve power and because of the difficulty of accurately sensing velocity at glider operating speeds" (Graver, 2005).

The vehicle glides horizontally by up to 20 degrees down from the horizontal plane and controls its trajectory by changing pitch angle and roll angle by displacing the center of gravity. To accomplish this, an internal actuator system changes the location of the center of gravity longitudinally and laterally by moving a weight. The vehicle has no external communication ability. It has a 3-liter dry pay load space for scientific measurement devices. It consists of a 1/2 ellipse shaped front cap, a cylindrical pressure hull, a corn shape tail cap

[3] No technical data of this vehicle has been released at printing.

with a vertical stabilizing fin, a pair of wings, tail wings and various electronic devices, i.e., a depth sensor, a gravity sensor, a magnetic sensor, two CPUs, interface boards and two actuators to trim and roll. A ranging sensor, a velocity sensor, a drop ballast system, a tail angle trigger and a transponder are fitted in the front and the tail caps (Kawaguchi et al., 1993).

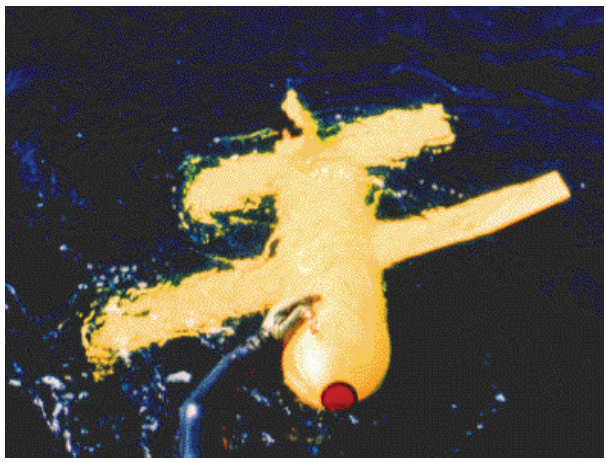


Fig. 12. ALBAC Glider (Kawaguchi et al., 1993)

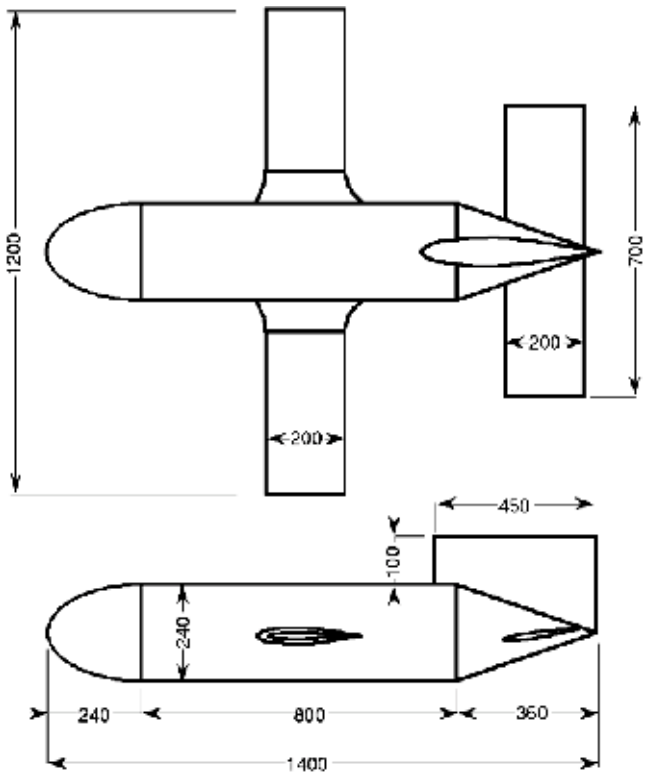


Fig. 13. ALBAC Glider Schematics (Kawaguchi et al., 1993)

Hybrid AUV-Powered Gliders
AUV-Powered-Glider

Another glider under development is a hybrid, which is designed to travel under power, glide mode or both. This vehicle, under development at Florida Institute of Technology, Melbourne Florida, is being designed to obtain water samples, make photographic/video images of specimens in the water column and specify the environmental characteristics of the data field. Furthermore, it is expected to possess a wide array of traditional oceanographic instruments that can be used by the vehicle's control system to make mission/navigational changes.

The vehicle's ability to obtain specimen/water samples and photographs directly affects the design of the vehicle more than the addition of oceanographic instruments. Water samples are to be collected using a series of small automatically closing specimen bottles, and two digital cameras are used to document what is floating through the water column.

The AUV-Powered Glider was design using the following parameters:

- Mission applications to 6000-meter ocean depths.
- Modular design: to ship easily in small boxes and to have interchangeable scientific modules.
- Quick assembly & disassembly of AUV components.
- Easy battery access for replacement and recharging during missions.
- Reasonable space for scientific & instrument payload.
- Capable of landing

Unlike torpedo-shaped survey AUVs, the structure of the AUV-Powered Glider has a rectangular frame that is approximately 1.5 by 2-meters square. Figure 14 shows an overview of an AUV-Powered Glider prototype with the main components.

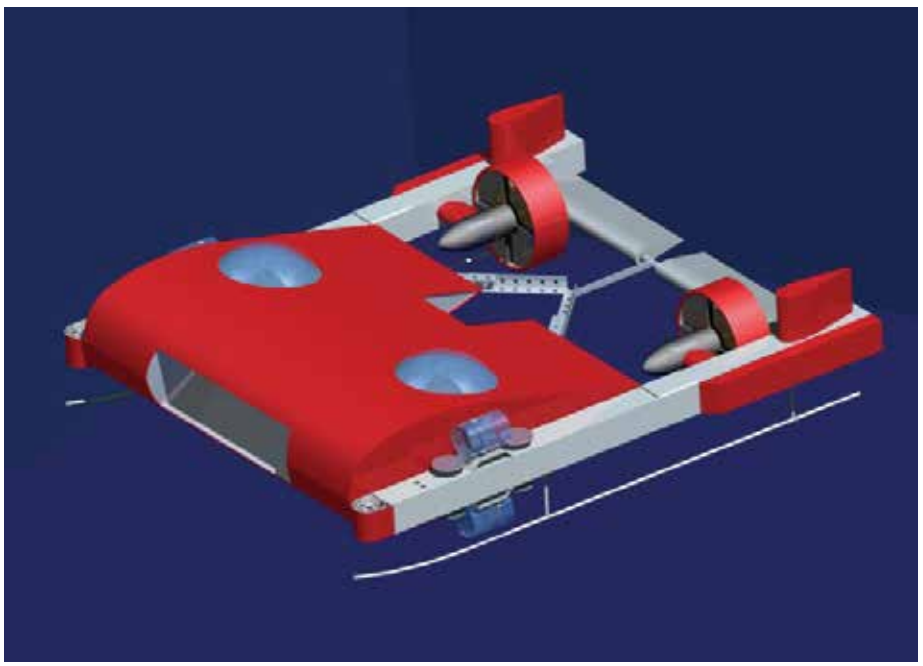


Fig. 14. AUV-POWERED GLIDER Prototype Overview

The vehicle is designed for easy assembly and disassembly, with easy access to the batteries and the two 17-inch diameter, 3/8-inch-thick vehicle control system and scientific pressure housings. The objective was to use cost-effective solutions to keep the overall budget of the vehicle reasonable. The version shown in figure 14 is for marine biologists, biological oceanographers and other scientists needing samples and photographs of organisms in the water column.

The main vehicle specifications for the AUV are:

- Dry weight: 293 kg (without instruments and drop weight system)
- Length: 1.93 m
- Width: 1.59 m, at flares 1.69 m
- Max. Height: 0.58 m
- Displacement: 379 kg
- Maximum depth: 4000 m
- Glass pressure housing depth: 6000 m

The AUV-Powered Glider is equipped with two 12-Volt longitudinal and two 12-Volt DC-brushless vertical thrusters mounted on the forward two corners of the frame.

- Longitudinal thrusters: asynchronous 3-phased, oil-filled design.
- Optimum running speed of 2-knots.
- Estimated power usage for the two thrusters at 2-knots, 12-Volts and 5-Amps = 50-Watts for each thruster.
- Vertical thrusters: Elcom ST N2312, coil-type 3-phase wye-wound, low speed, low operating voltage and high torque (K_t=5.30), 12-Volt DC-brushless motors from DC-brushless thrusters, are typically run up to 75% thrust and draw a total of 1.0-Amp for very short periods of time (e.g., one minute to raise the vehicle's bow from the ground in cases where the vehicle has landed).

Active Buoyancy Control - is used to make the vehicle's buoyancy either slightly positive or negative allowing the vehicle to glide up and down the water column in a saw-tooth pattern. The speed of the ascent or descent in glide mode depends on the buoyancy and glide angle and whether the vehicle is under power. The vehicle can be under power at any time, but energy consumption is high since the motors use more energy than any individual system on the AUV. A simple drop weight / drop float system is integrated currently for rapid prototype development allowing the vehicle 10 glide cycles. The design and development of a deep water buoyancy system is a primary task for future development of this vehicle.

Active Trim Control - is used to actively to control and stabilize the vehicle's trim. For example, when the buoyancy system has an unbalanced configuration (e.g., too much positive or negative buoyancy on one side) or when something foreign is tangled with the vehicle such as seaweed, the active trim control would attempt to align the vehicle. This control is handled by the rear control rudders and flaps. An automatic trim system using liquid mercury is under investigation that is similar to the trim systems on airplanes.

Fluid Intake Channel - at the front of the vehicle focuses water and organisms from in front of the vehicle through the channel. Two camera systems document what passes through the channel: one mounted so the photos are made from the side of the channel; the other mounted facing directly into the channel. An optional mesh can be mounted in front of the camera to collect organisms over a specified distance. The vehicle would reverse direction to wash already documented samples from the screen using the vehicle's thrusters.

Sample Taking - is made through a limited number of small sample chambers mounted along the external frame allowing the scientist to obtain permanent samples of the water and biological organisms. The sample chamber is opened and closed by servo motors at pre-set times.
Communication - is via a 802.11b Wireless Ethernet (WLAN) card between the AUV and a host PC allowing wireless communications with the AUV while at the surface and via radio through a MaxStream 9Xstream-PKG-R low-speed, half-duplex radio modem, with an extended range at sea: 7 miles (11km). Information concerning the MaxStream can be found at: (MaxStream, Inc., http://www.maxstream. net/).
Navigation and Absolute Positioning - is made with a Spartan Electronics SP3000D digital compass, depth gage and speed vector/altitude generated by a Doppler Velocity Log (DVL) for dead reckoning. Like any integrating process, dead reckoning accumulates errors and requires periodic fixes to cancel resulting drift. This is done by GPS during surface navigation. Collision control is through two UA-2 altimeters from J.W. Fishers Mfg., Inc. The altimeters have the pulse generation and return detection circuitry potted into the transducer and return the information to the computer via a RS232 connection. The UA-2 altimeters provide height over ground and the distance to an object in front of the vehicle up to 100 feet (30 meters) at 200 kHz. An inertial measurement unit (IMU) will measure the vehicle's acceleration and will determine the vehicle's position while underwater. The position will be verified by GPS when the vehicle is on the surface.
Control System and Supervision (See Figure 15) - algorithms manage the entire vehicle with a combination of a traditional feedback system and an under-development neural-network control system is used standard grid pattern surveys and chemical or physical trace mapping.
Sterne Hybrid Glider
The Sterne glider, developed at Ecole Nationale Superieure D'Ingenieurs in Brest, France is a hybrid glider having both a glider (buoyancy) and thruster mode. The 4.5 m long, 0.6 m in diameter, 900 kg in mass vehicle has buoyancy control and a thruster for forward propulsion and capable of gliding at 1.3 m/s.
The Sterne is designed to conduct surveys by gliding or by flying level using its thruster, which when powered has the range of an estimated 120 miles with an estimated speed of 3.5 knots (1.8 m/s). The vehicle has 2.5 knots (1.3 m/s) when gliding. It has two fixed wings two actuated horizontal tail fins and a vertical tail with rudder and moves a battery pack to control pitch (Graver, 2005).

5. Scientific sensors

An autonomous oceanographic data acquisition vehicle/glider that is usable by a wide range of scientists must be able to accommodate many different scientific instrumentation configurations, be capable of collecting specimens and be able to perform the missions as specified. Sensor packages are instrumental to a vehicle. Slocum, Spray, Seaglider and WaveGlider are too small for use with many types of instruments. Additionally, the saw-tooth glide pattern is not optimal for certain types of data collection such as Sidescan sonar. Only larger hybrid vehicles can make full use of all instrument types. Unfortunately, this forces the need of larger vessels and more manpower to deploy and recover these vehicles. Some of the instruments used on autonomous underwater vehicles that are rated down to 6000 meters are: **Sidescan sonar**; **Falmouth Scientific NXIC CTD** (a fully integrated

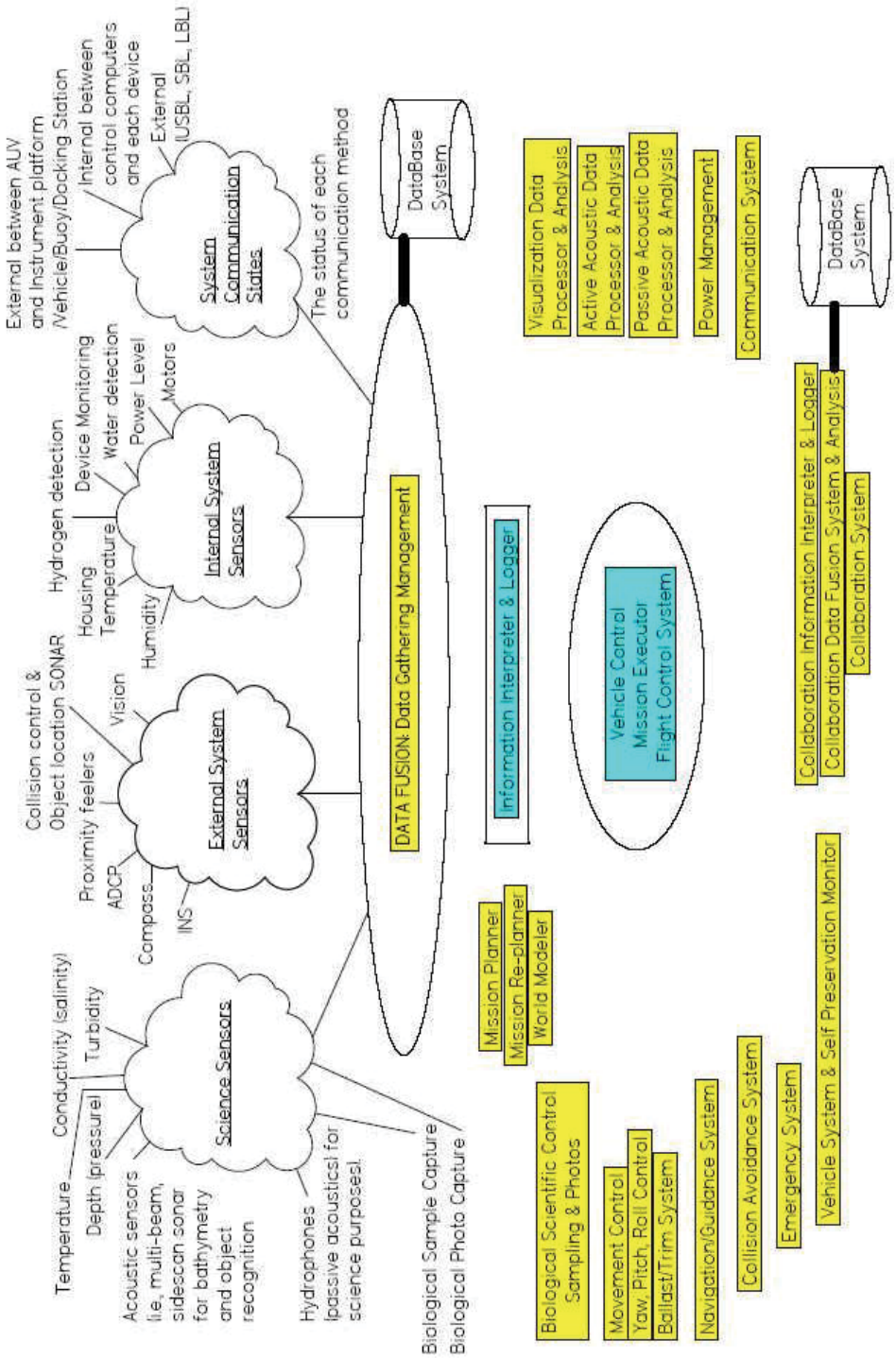


Fig. 15. AUV-Powered Glider's Autonomous Underwater Vehicle Systems Diagram

instrument platform (compact, robust and equipped with fully integrated conductivity, temperature, and depth sensors) with battery-power, internal data logging and external sensor input capability. It is designed to meet the demands of open-ocean, estuarine and fresh water environmental monitoring. It can be operated to a depth of 7000 meters and data may be stored on the internal storage memory or transmitted in real time via a serial interface (information available at http://www.falmouth.com/).); **Chelsea AQUA*tracka* III** (a compact, lightweight, submersible fluorimeter for the detection of chlorophyll-a, dye tracing or turbidity that when connected to the CTD Sensor provides measured values of chlorophyll, rhodamine, amido rhodamine, fluorescein. The AQUA*tracka* III is designed for depth up to 6000-meters. Applications: chlorophyll-a and other fluorophor detection, rhodamine and fluorescein dye tracing, particle concentration by light scattering, profiling, towed, moored or ROV deployment, pollution monitoring, bio-geochemical oceanography, and hydrothermal vent studies. This instrument can sense chemical fluorescence or light scatter in the visible and near infrared (400 to 800-nm). Versatility is achieved by the selection of appropriate optical narrow bandpass filters to match the excitation and emission wavelengths of the fluorophor, e.g., chlorophyll-a, rhodamine or fluorescein. It may be configured as a nephelometer by using the same bandpass filters for both excitation and emission.), **UV-VIS Spectrometer**, **video cameras** (provide high-resolution video or photo data that can be stored via a frame grabber to the integrated hard disk. The image data will be used, among other things, to qualify the initiated measurement locations offline and therefore document the measurement procedure), and **acoustic hydrophones**.

6. Future

As autonomous vehicles are developed to take on more responsibilities, program algorithms will be developed to accommodate these tasks. Currently, as mentioned in the WaveGlider section, new distributed on-board collaborative autonomous vehicle control programs are being developed that will enable an individual vehicle to coordinate and control multiple vehicles. This technique enables "swarm" capabilities among multiple vehicles. With on-board collaborative control, the vehicles operate as a group, functioning together as a "swarm." The swarm processes and communicates relevant information allowing individual vehicles and the entire swarm (i.e., group) to change direction, autonomously, in response to sensor inputs. This control is one of the primary research initiatives by the military for unmanned vehicle control in the air, on the ground and underwater. The concept of swarming is also useful to science for the sampling of entire regions for a specific organism, substance or phenomenon. The control of an autonomous underwater vehicle whether powered or a glider will also in the future utilize some combination of traditional (Figure 16) and neural network (Figure 17) navigation system that uses Kalman filters[4] to control the AUV.

One of the requirements for a long duration, autonomous underwater vehicle, is the need for a robust, fault tolerant, navigation system. In addition to the robustness issue, there are core issues of nonlinear control as they pertain to maneuverability and sea keeping. In both issues, neural networks offer very promising solutions. For example, the calculation of the

[4] A Kalman filter is a recursive filter estimating the state of a dynamic system. It is especially useful for handling incomplete or noisy measurements.

distances and the relative velocities will be by the use of the positioning data as well as by measuring inertial sensor data. In order to increase the reliability of the data, a reconciliation of both processes must be accomplished accurately and efficiently. The coordination of the target trajectories of the AUVs can give further important information for the positioning prognosis.

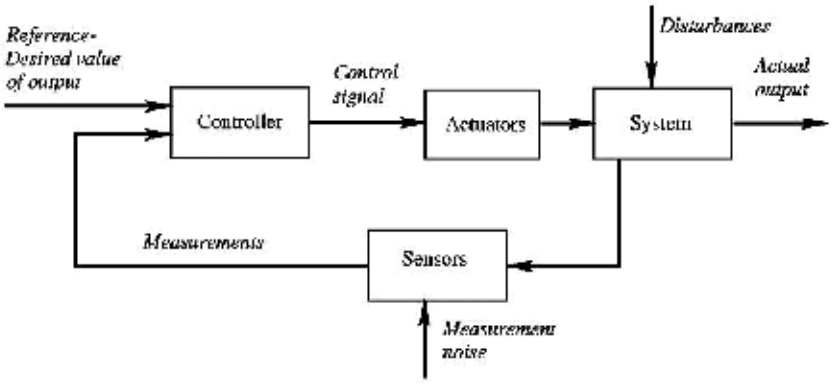


Fig. 16. Traditional Feedback System with Sensors

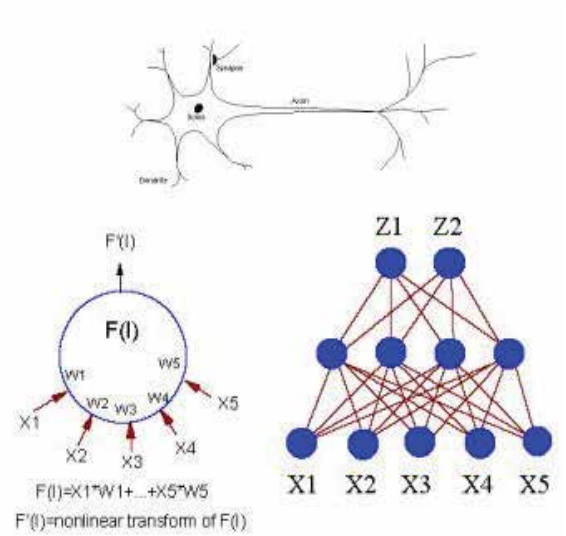


Fig. 17. The neural networks for control systems are based on human brain structure. The networks consist of artificial neurons, and each neuron is connected to other neurons through weights.

Current thruster powered commercial AUV systems use a combination of internal inertial, compass, and accelerometer sensors, in conjunction with external active acoustic triangulation methods (LBL, SBL, USBL)[5]. These have met with some success for applications of cable following, standard grid surveying, search and rescue, or signal

[5] LBL – Long Baseline, SBL – Short Baseline, USBL – Ultra Short Baseline.

following. But in each of these cases, the system is unable to respond to a) abrupt changes in external environment, b) system damage, c) uncertain or indeterminate data input. In these areas, some scattered research on the use of neural networks has been performed with success, addressing specifically the fault tolerance, docking, and ranging issues. For example, Wilson (Wilson, 1995) successfully evaluated the use of a neural network for a spaceship application providing robust navigation despite thruster failure. Most of the work in this area has been in spacecraft, but the work is directly applicable to underwater and surface vehicles. In most of the cases, a back propagating network is applied using position, rotation, or acceleration error as the training tool. In each case, changes to the vessel control system itself or in the external environment (displacement forces) causes the system to update its training, which in turn prompts it to compensate for the change in forces. Ship navigation has been evaluated using neural network based adaptive critic designs. For autonomous underwater vehicle (AUV) control, a neural network has been modeled at the University of Hawaii for the problem of depth gradient descent only. In each case, the results were very positive, indicating that if generalized, a full neural network system could provide robust navigation for an AUV.

In addition to the constituent issues above, there are many problems these vehicles are only beginning to address. Examples of these might include: search for environmental pollutants; search and analyze biological systems; locate and identify artificial acoustic sources; long term scanning for physical, biological, or chemical subjects of interest; non-inertial navigation.

The final step in the process is to use the processed multi-sensory data from the pattern recognition and data classification modules to provide control inputs for the navigation system. Thus, the system would then be able to track and monitor targets as listed above. In this phase, a simple feedback of neural network outputs will be sent to the control processor algorithm. The power of the neural network paradigm is the ability of the system to integrate the sensor input from a variety of sources into multi-sensory patterns, that is, acoustic with salinity, temperature and pressure, spectrographic with temperature, etc. But instead of traditional analytical methods where the individual datasets are correlated one by one, the neural network will be able to search for patterns in all sets together.

7. Conclusion

Autonomous Underwater Vehicles are only now being marketed as robust commercial vehicles for many industries, and of these vehicles underwater gliders are becoming the new tool for oceanographers. Satellites have provided scientists and marine specialists with measurements of the sea surface such as temperature since the late 1970s, and data via subsurface oceanographic moorings since the 1950's. As stated by David Smeed of the National Oceanography Centre, Southampton, England, *that* "gliders are one of the technological developments that are changing the way we observe the ocean and it is very exciting for us to be at the forefront of their application in ocean and climate science" (Douglas, 2008).

The Southampton team deployed a Slocum Glider on the 16th of September 2008 in the Eastern Atlantic (launched from the Canary Islands with the co-operation of the Instituto Canario de Ciencias Marinas (the Canarian Institute of Marine Science) with the aim of determining the interaction between oceans and climate and the intent to improve the ability of the scientist to detect signs of rapid climate change. That vehicle is expected to

travel 2,300 km over 90 days with a minimum of 1,000 profiles collecting temperature, conductivity (salinity), depth and current in its 1,000 meter depth range. The data retrieved will be made available to "the 'Rapid-Watch' program that monitors the meridional overturning circulation of the Atlantic. Also known as the 'Atlantic heat conveyor' this is the system of ocean currents that transports heat polewards, thereby influencing European climate" (Douglas, 2008).

The Rapid-Watch program is tasked to observe the Atlantic through 2014 with oceanographic moorings, ship observations and now autonomous underwater gliders. As stated by David Smeed, "the Rapid-Watch program is teaching us a great deal about how to monitor and evaluate changes in the ocean and climate. Underwater gliders are going to expand our capability to make these important measurements and enable us to get the data we need more efficiently."

Another important initiative is the "**The European Gliding Observatories (EGO) initiative,"** which is composed of oceanography teams from France, Germany, Italy, Norway, Spain, and the United Kingdom (but not restricted to European partners only) who are interested in developing the use of gliders for ocean observations throughout the world. More information concerning this initiative and becoming a member can be found at < https://www.locean-ipsl.upmc.fr/gliders/EGO/ >

8. References

APL (2007). The Applied Physics Laboratory Biennial 2007 Report, College of Ocean and Fishery Sciences, University of Washington.

APL (2008). Applied Research Laboratory, Seaglider Fabrication Center, University of Washington, Seattle, WA. Retrieved on 11 September 2008, <http://www.apl.washington.edu/projects/seaglider/specifications.html>

Argos (2008). Retrieved on 8 September 2008. <http://www.argo.ucsd.edu/ FrAbout_Argo.html>

AWI (2008a). Alfred Wegener Institute (AWI), Bremerhaven, Germany. Retrieved 11 September 2008. <http://www.awi.de/en/discover/click_learn/interactive/ice_tour/krill/>

AWI (2008b). Alfred Wegener Institute (AWI), Bremerhaven, Germany. Retrieved on 11 September 2008. <http://www.awi.de/en/research/research_divisions/biosciences/biological_oceanography/research_themes/zooplankton/ecology_and_ecophysiology_of_antarctic_krill_and_zooplankton/ >

Creed, E.L.; Mudgal, C.; Glenn, S.M.; Schofield, O.M.; Jones, C.P.; Webb, D.C., Oct. 2002. "*Using a fleet of slocum battery gliders in a regional scale coastal ocean observatory," Oceans '02 MTS/IEEE*, Volume 1, pp. 320 - 324.

D'Spain, G.L.; Jenkins, S.A.; Zimmerman, R.; Luby, J.C. and Thode, A.M. (2005). Underwater acoustic measurements with the Liberdade/X-Ray flying wing glider. *J. of the Acoustical Society of America*, 117, 4, pp. 2624.

D'Spain, G.L.; Zimmerman, R.; Jenkins, S.A.; Luby, J.C.; and Brodsky, P. (2007) "Underwater acoustic measurements with a flying wing glider," *J. Acoust. Soc. Am.*, Vol. 121, No. 5, Pt. 2, May 2007, pp. 3107.

Davis, R. E., Webb, D. C., Regier, L. A., and Dufour, J. (1992). "The autonomous Lagrangian circulation explorer (ALACE)," *J. Atmos. Oceanic Technol.*, vol. 9, pp. 264–285, 1992.

Douglas, M., (2008). "*Glider joins Rapid-Watch Ocean Monitoring Program," Marine Technology Reporter,* 15 September 2008, retrieved on 15 September 2008. <http://www.seadiscovery.com/mt/mtStories.aspx?ShowStrory=1026022051>

Eriksen, C.C.; Osse, T.J.; Light, R.D.; Wen, T.; Lehman, T.W.; Sabin, P.L.; Ballard, J.W.; Chiodi, A.M. *(2001). "Seaglider: A long range autonomous underwater vehicle for oceanographic research," IEEE J. Oceanic Engin., Volume 26, Issue 4, pp. 424-436.*

Frey, C. L. (2008). Conversations with Charles (Lee) Frey, 11 September 2008, Ocean Engineer at Harbor Branch Oceanographic Institute, Fort Pierce, Florida.

Graver, J.G. (2005) "Underwater Gliders: Dynamics, Control and Design," Dissertation, Princeton University, Department of Mechanical and Aerospace Engineering, May, 2005.

Griffiths, G., Ed., Davis, R. E., C. C. Eriksen, and C. P. Jones, (2002). Autonomous buoyancy-driven underwater gliders, In: Technology and Applications of Autonomous Underwater Vehicles, Taylor and Francis, London.

Herring, D., Chief Editor (2007a), NASA's Earth Observatory, "What are Phytoplankton?" Retrieved on 11 September 2008. <http://earthobservatory.nasa.gov/Library/Phytoplankton/>

Herring, D., Chief Editor, (2007b) NASA's Earth Observatory. Retrieved on 11 September 2008. <http://earthobservatory.nasa.gov/Observatory/Datasets/chloro.ocean.html>

Kawaguchi, K.; Ura, T.; Tomoda, Y.; Kobayashi, Y. (1993) "Development and Sea Trials of a Shuttle Type AUV "ALBAC"", *Proc. 8th. Intn. Symp. on Unmanned Untethered Submersible Techonology*, Durham, Sep. 1993, pp.7-13.

Mote (2007). Mote Marine Laboratory, Sarasota FL, "ABOUT RED TIDE …." Retrieved on 11 September 2008. <http://isurus.mote.org/~mhenry/WREDTIDE.phtml >

ONR (2006). "Liberdade XRAY Advanced Underwater Glider," ONR press release, retrieved on 15 September 2008. <https://www.onr.navy.mil/media/extra/fact_sheets/advanced_underwater_glider.pdf>

Osse, T.J.; Lee, T.J. (2007). "*Composite Pressure Hulls for Autonomous Underwater Vehicles," Oceans 2007*, Sept. 29 2007-Oct. 4 2007, pp. 1–14.

Osse, T.J.; Eriksen, C.C. (2007). "*The Deepglider: A Full Ocean Depth Glider for Oceanographic Research," Oceans 2007*, Sept. 29 2007-Oct. 4 2007, pp. 1–12.

Seaglider (2008). Retrieved on 11 September 2008, <http://www.apl.washington.edu/projects/seaglider/summary.html>

Sherman, J., Davis, R. E., Owens, W. B., and Valdes, J. (2001). "*The autonomous underwater glider Spray," IEEE* J. Oceanic Engin., Volume 26, Issue 4, Oct. 2001, pp. 437-446.

Spray (2008a) Scripps Institute of Oceanography, University of California, San Diego Instrument Development Group, La Jolla, CA. Retrieved on 10 September 2008, http://spray.ucsd.edu/pic/spray_desc.pdf

Spray (2008b) Retrieved on 11 September 2008. <http://spray.ucsd.edu/>

Webb, D. C., Simonetti, P. J., Jones, C.P. (2001). "SLOCUM: an underwater glider propelled by environmental energy," *IEEE J. Oceanic Engin.*, Volume 26, Issue 4, Oct. 2001, pp. 447 – 452.

Webbresearch (2008a) Retrieved on 9 September 2008, <http://www.webbresearch.com /electric_glider.htm>

Webbresearch (2008b) Retrieved on 9 September 2008, <http://www.webbresearch.com /thermal_glider.htm>

Wilson, E. (1995). "Experiments in Neural Network Control of a Free Flying Space Robot," NASA Contractor Report, CR-206048, 1995.

27

Cooperative Acoustic Navigation Scheme for Heterogenous Autonomous Underwater Vehicles

Xianbo Xiang[1,2], Lionel Lapierre[1], Bruno Jouvencel[1],
Guohua Xu[2] and Xinhan Huang[2]
[1]*CNRS - LIRMM – University of Montpellier II*
[2]*Huazhong University of Science and Technology*
[1]*France,*
[2]*China*

1. Introduction

Oceans cover 71% of the earth's surface and contribute the largest reservoir of life on the earth. With more and more concern about the abounding and valuable ocean resources, these years have witnessed a remarkable growth in the wide range of underwater commercial activities for ocean survey, especially focusing on undersea exploration and exploitation, and even extensively for salvage operations related with disastrous accidents occurred undersea (Lapierre, 2006).

There are three main kinds of vehicles recruited for underwater activities. Manned Submersibles and Manned Underwater Vehicles with good abilities of directly manoeuvring and in-situ observation, have been widely utilized in commercial activity and scientific research, and reached the zenith in the late 1960s and early 1970s. However, this critical systems with vital importance of crew aboard and complex handling system significantly cost so much. Then, Remotely Operated Vehicles (ROVs) still with human in the loop but not in the vehicle are successful substitutes, being low-cost vehicles piloting in deep water greater than 1000ft. Today, ROV becomes a well-established technology frequently used in the offshore industry, most notably in the commercial offshore oil and gas, nuclear, pipeline and cable industries. Nevertheless, the long umbilical cable, linked with the mother ship, greatly inhibits the speed of the ROV, requiring the mother ship equipped with deck gear capable of winding up this cable and significantly restricting ship movement while deployed. More recently, with the development of advanced underwater technology, Autonomous Underwater Vehicles (AUVs) are steadily becoming the next significative step in ocean exploration due to their freedom from the constraints of an umbilical cable. Nowadays there has been gradually growth in the AUV industry worldwide which would be on an unprecedented scale and AUVs will carry out interventions in undersea structures in the future (Whitcomb, 2000). Moreover, recent applications using Intervention Autonomous Underwater Vehicles (IAUVs), have demonstrated the feasibility of autonomous underwater manipulations (Xu et al., 2007), controlled via acoustic links, thus removing the parasite effects of the umbilical cable (http://www.freesubnet.eu). With

further research results and technological advances, AUVs have the potential for supplementing or even substituting ROVs for deep water operations, and AUVs in a team hold considerable potential for challenging scientific and commercial mission at sea.

As a group of coordinated multiple robots dealing with tasks provides flexibility, robustness and efficiency beyond what is possible with single robot, there is one attractive scenario for underwater activities--the AUV team concept, which could be a mix of several low-cost specific purpose AUVs, guided and controlled by one or two higher cost AUVs (Xiang et al., 2008). The employment of multiple AUVs has significant advantages for both military and commercial applications (Bourgeois et al., 1999). A team of underwater vehicles could survey large ocean areas more rapidly and economically than that could be accomplished with a single AUV or ship (McDowell et al., 2002). The key point to the operation of AUVs is the availability of accurate navigation and positioning systems, which provide the measurement of the angular and linear positions of each underwater vehicle in the team and is therefore crucial to control and stabilize the platform. Unfortunately, one of the major problems that prevents the commercial application of AUVs, or at least mitigate their efficiency, is just that of vehicle navigation. On board navigation systems, as inertial navigation systems (INS), can not maintain the requested accuracy over the long time vehicle manoeuvring and are highly expensive as well as the inconvenient calibration for different AUV systems due to its vehicle-specific characteristic (Caiti et al., 1999).

There are several positioning and navigation systems currently employed by AUVs researchers. The traditional acoustic navigation methods will be reviewed in section 2, and the main non-acoustic approach, which is also a dominant approach for AUVs, is combining a GPS receiver and an INS in one AUV (INS/GPS). That is, the vehicle mainly depends on the INS to be navigated, but periodically comes to surface to receive the GPS signal and to recalibrate the INS (Yun et al., 1999).When one group of AUVs is traveling to the area of interest, inter-vessel communications could also be used to provide the information of position and navigation, and then the team of AUVs relies on machine learning techniques for creation and maintenance of loose formation. But there is an important assumption that still at least one vehicle has an accurate positioning system on board, typically with the INS combined with GPS. That means at least one of the AUVs must periodically come to the surface to calibrate the position which would severely disturb or even deteriorate the whole strategy of the team coordination and formation, besides the unwanted energy consumed to heave up to the surface and the high cost of INS.

Accounting for the disadvantage of currently positioning and navigation approach for coordinated AUVs team mentioned above, another promising scheme is the heterogenous autonomous vehicle team concept to overcome the navigation problem, which would be a mix of several low-cost specific purpose vehicles which typically are AUVs, guided and controlled by one or two higher cost control vessels which typically are ASVs. Benefited from the underwater GPS concept combining the DGPS technology, a dedicated novel cooperative underwater acoustic navigation approach is suitable for this heterogenous vehicle team. The central control ASV can get high precise positions of AUVs without INS/GPS on board, allocates the waypoints to the AUVs as well as provide the navigation information via acoustic modem and also move above the central of mass of the AUVs, so that the whole team with heterogenous vehicles could conveniently implement the coordinated search or rescue scenario as a whole (Xiang et al., 2007).

The rest of this chapter is organized as follows. In section 2 the traditional underwater acoustic navigation system and the underwater GPS concept are reviewed, and the hardware implementation of DGPS intelligent sonobuoys as well as the novel cooperative navigation architecture for heterogeneous autonomous vehicle is presented in section 3. Section 4 includes a detailed description of the cooperative navigation algorithm for coordinated underwater vehicles. Section 5 provides the simulation results of the acoustic navigation. Section 6 draws conclusions. Section 7 makes acknowledgement for the support from co-authors and related scientific research projects.

2. Traditional navigation methods for AUVs

In this section, two kinds of navigation systems currently employed by AUVs will be reviewed here. One is acoustic navigation system, the other is non-acoustic navigation system, especially the underwater GPS navigation system.

2.1 Acoustic navigation

The simple transition from available navigation techniques based on electromagnetic signals for mobile robots or flying robots to underwater vehicles, is not applicable due to the peculiarities and constraints of the underwater environment, as the electromagnetic signals do not penetrate below the sea surface. The good propagation characteristics of sound waves in water makes acoustic positioning and navigation as a feasible candidate, and the related study of the implications of such methodology for the underwater vehicles has been conducted for a long time.

As fig. 1 illustrated, classic acoustic approaches for underwater vehicle positioning, include Long Baseline (LBL), Short Baseline (SBL), Ultra-short Baseline (USBL) Systems, and Long & Ultra Short Baseline (LUSBL), etc.

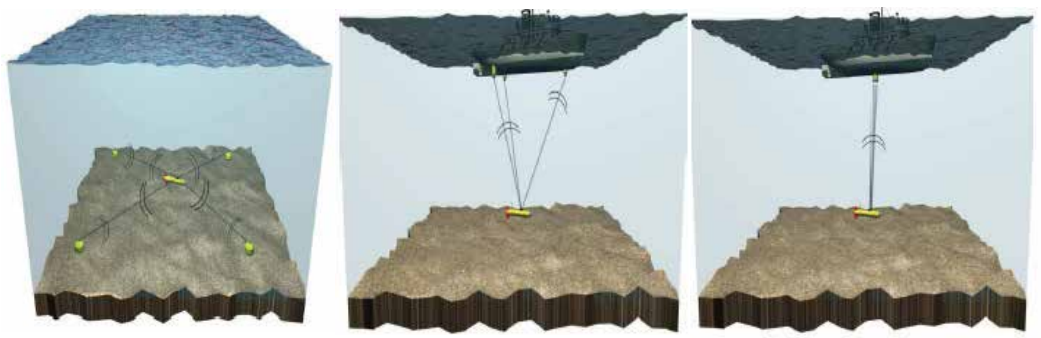


Fig. 1. Classic underwater acoustic positioning systems: LBL system(left), SBL system(central), USBL system(right).

The application and performance of this challenging area have been investigated by many researchers (Vickery, 1998). In the LBL case, a set of acoustic transponders is pre-deployed on the seafloor with the geometry of interested vehicles centered. The vehicle position is achieved by the basis of the acoustic signal returns detected by the transponders with the required accuracy (Collin, 2000). In the SBL side, a dedicated ship follows the underwater vehicle at short range with a set of three hydrophones to determine the AUV position, and the AUV can also get its absolute position via the bidirectional communication among the

AUV and the mother ship (Storkensen, 1998). USBL systems are very similar to SBL principles except that the transducers are built into a single transceiver assembly or an array of transducer elements in a single transceiver. The distances are measured as they are in an SBL system but the time differences are replaced by the "time-phase" of the signal in each element with respect to a reference in the receiver. The "time-phase differences" between transducer elements are computed by subtraction and then the system is equivalent to an SBL system. The LUSBL system is a special case of a USBL system. It utilizes USBL hardware in a configuration similar to the one described for the LBL system. Range and bearing in an LUSBL system are still measured as described for a basic USBL system. However, because a larger number of beacons are deployed on the seabed, a considerable improvement in accuracy may be achieved.
Although all these classic acoustic methods have been used for a long time, there are still some disadvantages existed in practical utilization. LBL systems require long time with associated costs for deployment and comprehensive calibration at each deployment. SBL systems is installed on a dedicated ship so that they are in poor signal to noise ratio due to ship's self noise and the accuracy of acoustic positioning can only be achieved in calm weather and without ship motion which also lies on USBL and LUSBL systems.

2.2 Underwater GPS navigation

As mentioned above, traditional acoustic solutions for AUV navigation present some installation, calibration constraints and operational limitations. Their performances may be over estimated and in some cases not fully satisfying, and then non-acoustic solutions will be considered here.
The traditional non-acoustic approach, is a set of INS on board combining with a GPS receiver, which is also a dominant approach for AUVs. Due to the accumulated error from INS, the AUV must periodically come to the surface to calibrate the position with the help of GPS. In the case of a team of multiple AUVs, at least one AUV, providing accurate navigation information for others, have to come to the surface for position calibration, which would deteriorate the whole strategy of the team coordination and formation, besides the extra energy consumed to come to the surface and high cost of INS set in each AUVs.
Since there some drawbacks of the non-acoustic INS combined with GPS navigation approach for coordinated underwater vehicles, we should seek alternative solutions. Unfortunately, it seems there is no way to directly utilize GPS for underwater navigation, as the electromagnetic signals do not penetrate below the sea surface making the GPS unsuitable for directly underwater navigation. However, more recently, several new ideas about underwater "reproducing" GPS have been proposed in order to improve the accuracy of underwater positioning and navigation, making such system easily used. The ideas of "reproducing" the GPS in the underwater environment which getting the merits of both non-acoustic and acoustic approaches, can be classified in three different groups summarized as follows.
The first type is so-called "false" underwater GPS. A GPS receiver mounted on a buoy is towed on the surface by the underwater targets such as underwater vehicles. A cable or fiber is used to send the GPS position to the underwater target. This technique does not give the true position of the target but the false position even in few tens of meters around the surface buoy, so that it is named as false underwater GPS.

The second type is a "direct" underwater GPS solution. In 1992, Youngberg inspired a direct transposition of GPS signal to underwater world. Acoustic waves but not radio electric signals, directly go from surface buoys replacing satellites to the underwater mobiles (receivers). Then, the underwater platform receives these acoustic messages from the buoys equipped GPS receptors and computes its own position locally. This solution has been presented by M. Youngberg of the US-AIR FORCE and patented (US Patent N°: 5.119.341).
The third type is very similar to the second type solution, but it is a "reverse" underwater GPS solution. This method has been recently investigated by Thomas (Thomas, 1998) and is available commercially: the so-called GPS Intelligent Buoy (GIB) system, developed by ACSA in 1999. This system is designed to track the position of an underwater target equipped with an acoustic emitter, by measuring the times of arrival of the acoustic signals at a set of surface buoys equipped with submerged hydrophones and GPS receivers. As fig. 2 illustrated, it is a standard GIB buoy system (http://www.underwater-gps.com). The minimum number of buoys deployed is 2 if there are only 2 unknown parameters X and Y, as the depth (Z) of the target could be known using a telemetry channel. Technology of Time, Frequency or pseudo-random code diversity could be employed when tracking multiple underwater targets.

Fig. 2. GIB standard buoys

As we can see, in the direct type 2 solution, the information fusion is done on board the vehicle which uses downwards acoustic flow of data. In the reverse type 3 solution, the fusion is done on the remote control station which uses upwards acoustic flow of data.

3. Cooperative acoustic navigation scheme for AUVs

Inspiration of new techniques allows the underwater usage of DGPS, that is "Underwater DGPS" concept, to develop advanced underwater robotics system. Because of its special characteristics, the "reverse" type 3 is selected as the representative acoustic navigation system for heterogenous autonomous vehicles, and with which the heterogenous vehicles in a team could benefit a lot from the cooperative acoustic navigation. Before referring to the cooperative acoustic navigation scheme for heterogenous vehicles, the DGPS Intelligent Sonobuoy (DIS) navigation system will be firstly introduced here.

3.1 DIS navigation system

A prototype of such acoustic navigation system proposed in this chapter consists of a set of light surface buoys usually more than three of them. The physical characteristics of the buoys with acoustic module(hydrophone), in terms of size, weight and autonomy, will tend to those of a standard sonobuoy, with however the capability of micro-controller, processing of acoustic signals, local data storage, and online transmission by a radio modem. The most peculiar characteristic is a DGPS receiver integrated on board, the hydrophone and the ARM CPU is synchronized before deployment with the DGPS clock datum. One possible simplified schematic of a sonobuoy with DGPS is shown on fig. 3. We call it as a DGPS Intelligent Sonobuoy system.

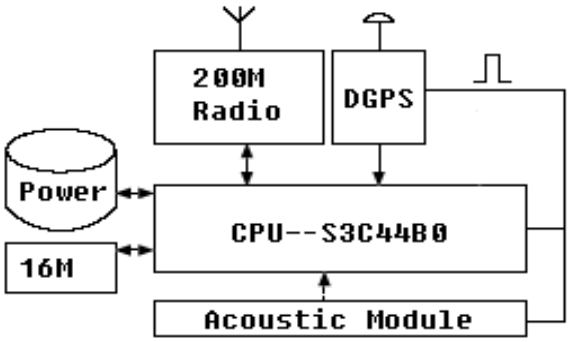


Fig. 3. Block scheme representing DGPS Intelligent Sonobuoy

The core of the DIS is ARM7 CPU named as S3C44B0. An extended 16M memory is for data storage as well as a data buffer. An acoustic module based on MOTOROLA DSP. A two way radio link with 200M Radio Frequence (RF) allows for communication within ranges up to 40/50 km. The DGPS board for 1us precision timing and localization of the buoy. The system was designed to be deployed switch off and auto-activate in order to avoid electronic equipment damage by severe deck and banging during deployment and recover.
The DIS system usually consists of four surface buoys which use the hydrophones to receive the acoustic impulses and record their times of arrival (TOA). The pinger onboard the vehicle emits two successive acoustic pulses during each emission cycle so that the pinger/vehicle depth is proportional to the time delay between the two pulses. The ranges or distances between the buoys and the pinger/vehicle can be translated into the times of arrival, and the value of sound speed in the water is assumed to be known. The buoys transmit the information of the DGPS position and the time of arrivals to the a Autonomous Surface Vehicle as a central station via radio link, where the true position of underwater target is computed by triangulation or by using more sophisticated algorithms. However, the dropouts and outlier of the acoustic positioning system should be carefully dealt with due to acoustic path screening, partial system failure, and multipath effects. See (Vagannay et al., 1996)for a treatment and discussion on this challenging topic, and (Alcocer et al., 2004) also gives an intensive study to deal with the dropouts and outliers in real time.

3.2 Cooperative navigation architecture for heterogenous AUVs

With the underwater DGPS concept and the underwater acoustic navigation approach, the position of the AUV can be achieved with high precise given a set of range measurement

from the AUV to known sonobuoys locations. In the set-up adopted for vehicle positioning the underwater pinger of an AUV carries a high precision clock that is synchronized with those of the sonobuoys (and thus with DGPS) prior to AUVs deployment. The pinger emits an acoustic signal every T seconds, at precisely known instants of time. As the underwater positioning of AUVs is done on the remote control station which uses upwards acoustic flow of data, one natural strategy for the undersea exploration is to use an Autonomous Surface Vessels (ASV) to control multiple AUVs motivated by the "reverse" underwater GPS method. With this method, the ASV can conveniently get all of the AUVs' positions in a heterogenous coordinated AUV team, because the pre-deployed four sonobuoys as a set transmit the DGPS positions themselves and the TOA of AUVs in a team to the central control ASV via radio link, and then generate and allocate the waypoints to the AUVs to implement the coordinated search or rescue scenario. As fig. 4 illustrated, a typical description of such heterogenous vehicle team is a ASV as a central control vehicle combined with three types low-cost AUVs based on DIS to provide navigation information. One type of them may be a vehicle with side-scan sonar to build maps of the ocean bottom and get rough list of interested objects with on-board computer-aided detection and classification. It will be followed by another vehicle with forward-looking high resolution sonar or camera for further identification of the object. The last type of vehicle with a dexterous underwater manipulator maybe intended to perform some sampling inspection operations.

Fig. 4. DIS for coordinated control of heterogeneous vehicles

The ASV communicates with AUVs via 3~5 kHz acoustic modem for long range bi-directional communication. Communications from the ASV to the AUVs in the heterogeneous team will support navigation and control information. Control commands include performing wide-scale mapping, making detailed acquisition or further identification over targets, drilling and sampling of interested objects, and so on. Navigation messages include the waypoint generated by the ASV as well as the positioning information for the specific AUV which will be accurately provided by the acoustic modem. AUVs will

upload useful sensor information and compressed high resolution segments of data to the central ASV, and this function will be unnecessary if the vehicles are to be retrieved with all datum transferred on shore.
Without the INS/GPS onboard, the AUV with DIS need not periodically come up to the surface to calibrate their positions however seriously disturb the underwater vehicle team coordination and existed formation. In the side of underwater vehicles for navigation, just a good flux gate compass is the most important requirement, and a Doppler velocity sensor associated with a high quality attitude sensor can improve the overall system accuracy. From time to time, the AUVs will receive new heading settings via an acoustic link. The central control ASV will also move above the central of mass of the AUVs in order to keep a desirable distance between the ASV and AUVS to ensure the high quality acoustic communication. Thus the whole team with heterogenous vehicles could conveniently implement the coordinated search or rescue scenario as a whole base on the novel acoustic navigation scheme. Compared with traditional acoustic positioning methods, the high precision could also be conveniently achieved by the DIS without installation and calibration constraints and operational limitations. With this novel acoustic navigation scheme, it is possible for us to easily handle with heterogeneity in coordinated control task for underwater activities.

4. Cooperative navigation algorithm

Problem statement: Given the simultaneous information of sonobuoys positions and the TOAs at different sonobuoys, and assuming the measurement performed each time affected by a bounded error accounting for all the uncertainties, then the simultaneous Least Squared (LS) navigation algorithm could be proposed for cooperative navigation for heterogenous AUVs, and the observation equations could be made. The Taylor-series will give a Least Squared error solution to a set of simultaneous nonlinear algebraic position equations, and the position of AUV will be achieved via the iterative refinement scheme.
When Time of Arrivals from the pinger on-board the vehicle to the sonobuoys combined with the DGPS positions of the sonobuoys themselves are used to identify the position of an AUV, it is possible for the sonobuoys to get the positions of multiple underwater vehicles within different time slots, namely the time-multiplexed navigation. Then, the central control ASV can collect all the positions of underwater vehicles via radio link to sonobuoys. It gives a clear understanding that the navigation algorithm for acoustic navigation in coordinated underwater vehicles regresses to the navigation algorithm of a single target. The detailed algorithm will be shown as follows.
Consider an absolute earth fixed reference system {O}:={X, Y, Z} with the Z-axis pointing upward from the sea surface, and n sonobuoys(usually n is equal to four) at the sea surface with hydrophones at DGPS positions given by vector $P_i=[x_i, y_i, z_i]^T$; $i=1,2,\ldots n$. Let $P_0=[x\ y\ z]^T$ denote the position of one of the interested underwater vehicles with respect to the reference frame. The navigation problem considered in this chapter can then be concisely stated as follows: obtain estimates of the AUVs position based on information provided by the sonobuoys, which compute the flight time of the acoustic signals emitted periodically by a pinger installed on-board the underwater vehicle. It belongs to a passive acoustic navigation system but not an active navigation system according to (Freitag et al., 2001). Further let $f=[f_1...f_i]^T$; $i=1,2,\ldots n$, denote the ranges between the dedicated underwater vehicle and the sonobuoys.

Before formulating the measurement, there are three assumptions based on practical but also reasonable principle.

1. Each sonobuoy position will be known at differential GPS accuracy, and the uncertainty of the DGPS position can be treated as an additional uncertainty in the totally measurement.
2. The sonobuoys freely drift at the sea surface due to waves and current and are assumed with a much slower drifting speed compared with the instantaneous flight time of the acoustic signals, which means the sonobouy position is relatively static and the movement of the sonobuoy can be treated again as an additional uncertainty in the totally measurement.
3. The sound speed *c(z)* in the area of interest is assumed known and the sound speed is assumed to vary only with depth in the area of interest is considered in the most common situation. Although the different sea temperature do affect the sound speed in different water column, the slightly varied sound speed due to heterogeneous temperature in different current layers can be treated again as an additional uncertainty in the totally measurement.

With the simultaneous information of sonobuoys positions and the TOAs at different sonobuoys, the simultaneous Least Squared (LS) navigation algorithm can be followed as:
The *n* equations describing the distance between the unknown position of the AUV and the sonobuoy location are:

$$f_i = \| p_0 - p_i \| = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} \tag{1}$$

The underwater pinger carries a high precision clock that is synchronized with those of the sonobuoys with DGPS prior to AUV deployment. Assuming the measurement s_i of the one-way travel time of the emitted signal by the pinger of the underwater vehicle is available to the *i*th sonobuoy, the distances between the AUV and the sonobuoys can be simply computed as

$$f_i = s_i \cdot c(z) \tag{2}$$

The sound speed is not a constant and varies with depth of the sea. Practically the weighted average speed is used to evaluate the sound speed propagating under the sea where the profile of the sound speed is divided as *m* layers, is shown on fig. 5. Then the sound speed considered in the central control ASV can be defined as

$$c(z) = \frac{1}{H} \sum_{i=0}^{m-1} (c_i + c_{i+1}) \cdot (z_{i+1} - z_i) / 2 \tag{3}$$

Assuming the measurement performed each time affected by a bounded error e_i, which accounts for all the uncertainties, such as synchronization errors, noisy signals, DGPS accuracy, buoy/vehicle motion between transmission and reception of the same ping, multi-path effect, etc. Then the observation equations with uncertainties can be written as

$$\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} = s_i c(z) - e_i \tag{4}$$

As the Taylor-series estimation (also Gauss or Gauss-Newton interpolation) gives a Least Squared error solution to a set of simultaneous nonlinear algebraic position equations (Foy, 1976). If x_g, y_g, z_g are initial guesses of the true vehicle position, f_i can be expanded in a Taylor-series keeping only terms below second order as

$$f_{ig} + \frac{\partial f_{ig}}{\partial x}\delta x + \frac{\partial f_{ig}}{\partial y}\delta y + \frac{\partial f_{ig}}{\partial z}\delta z \cong s_i c(z) - e_i \tag{5}$$

Where

$$\delta x = x - x_g, \delta y = y - y_g, \delta z = z - z_g$$

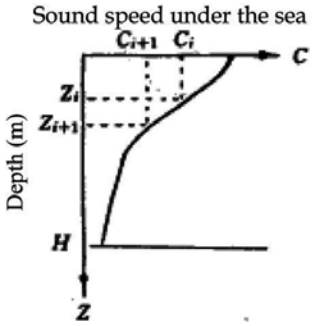


Fig. 5. Profile of the sound speed under the sea

Assuming errors e_i are stastistically distributed, the approximate relations of stacked equations (5) can be written as

$$A\delta \cong b - e \tag{6}$$

Solving (6) for δ while minimizing the least squared error with the terms in the sum weighted according to the covariances of the measurement errors gives

$$\delta = [A^T R^{-1} A]^{-1} A^T R^{-1} b \tag{7}$$

The matrixes and vectors are defined as:

$$\delta = \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}$$

$$A = \begin{bmatrix} \frac{x_g - x_1}{\rho_1} & \frac{y_g - y_1}{\rho_1} & \frac{z_g - z_1}{\rho_1} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{x_g - x_n}{\rho_n} & \frac{y_g - y_n}{\rho_n} & \frac{z_g - z_n}{\rho_n} \end{bmatrix}$$

$$b = \begin{bmatrix} s_1 c(z) - f_1 \\ \cdot \\ \cdot \\ s_n c(z) - f_n \end{bmatrix}$$

Where

$$\rho_i = \sqrt{(x_g - x_i)^2 + (x_g - y_i)^2 + (x_g - z_i)^2}$$

With the computed δ x, δ y, δ z replace

$$x_g = x_g + \delta x, y_g = y_g + \delta y, z_g = z_g + \delta z$$

in (6), and repeat the computation to get a new estimate of P_0=[x y z]^T. The iterations will have converged when δ x, δ y, δ z are essentially zero, and the position of AUV is achieved via the iterative refinement scheme. The solution derived above can be easily simplified to the case where the target undergoes motions in two dimensional space with fixed depth z.
Considering the noise covariance matrix R= σ^2 I (i.e., the noise terms e_i are independent with the same variance), the covariance matrix of the LS estimation errors in the corresponding position can be calculated as

$$C = \left[A^T R^{-1} A \right]^{-1} = \left[A^T A \right]^{-1} \sigma^2$$

The system is computationally simple requiring inversion of n (sonobuoys as reference stations) by 3 matrix and it usually converges in several iterations, but the linearized estimator is sensitive to the geometry of the situation. For instance, sonobuoys were placed very clear each other and the AUV is in very deep depth, then even small measurement errors of TOA will lead to large position estimation errors due to the occurred ill-conditioned matrix.
On the other hand, the solved depth of the underwater target is very inaccurate when depth is small compared to the distance between transponders. In this case, a pressure sensor integrated in the mobile's pinger is one solution, and the telemetry technique used is of an external freedom for depth accuracy as the time delay between two pulses is proportional to the depth.
Within different time slots, all of the dedicated AUVs in the coordinated team can be calculated with this high precise navigation algorithm based on intelligent sonobuoys with DGPS. Benefited from all of the vehicle positions, The ASV can generate and allocate the waypoints to the AUVs as well as providing the navigation information so that the coordinated control of heterogenous vehicles including the ASV and AUVs could be possible. At the same time, the ASV with the collected AUV positions will follow the AUVs and move above the central of mass of the underwater vehicles to keep the heterogenous team as a whole (Xiang[a] et al., 2007).

5. Simulation results

The performance of the cooperative navigation proposed here could be examined by computing the trajectory-tracking error with noisy range measurements, which was with very good accuracy. The acoustic navigation system was tested as follows.
It was supposed that there were four sonobuoys pre-deployed in a given square geometry of 10x10 Km in the X-Y plane, and the AUV of interesting in the coordinated team was in the depth of 500m. The sonobuoys were separately located at (0, 10000, 500), (0, -10000, 500), (10000, 10000, 500), and (10000, -10000, 500), and the AUV was locate at (0, 0, 0). The performance of the navigation method could be examined by computing the trajectory-

tracking error with noisy range measurements. Assume that the range measurements in four sonobuoys had a standard deviation of σ_1^2 =0.5, σ_2^2 =4, σ_3^2 =1, and σ_4^2 =0.2. Assume also that the errors were independent. Thus the covariance matrix of the errors was R=[*0.5 0 0 0; 0 4 0 0; 0 0 1 0;0 0 0 0.2*]^T. In fig.6, the solid track was from the true trajectory of one AUV in coordinated team moving in a spiral cord in 3D space or projected in 2D space, and the asterisks represented the corresponding tracking position estimates for the acoustic navigation system, which were with very good accuracy. Using the simultaneous LS positioning algorithm, the estimates were typically within centi-metric accuracy in northing(*X*-axis) and easting direction(*Y*-axis) of the true position. The mean length of the overall errors was 0.014171m and 0.0013966m respectively in *X*-axis and *Y*-axis, and the mean length of correction vector values was 0.2215m. The small errors give accurate measurement of the AUV position, but we should also be aware of that the actual error might be different due to the variety of noise measurements and the DGPS accuracy.

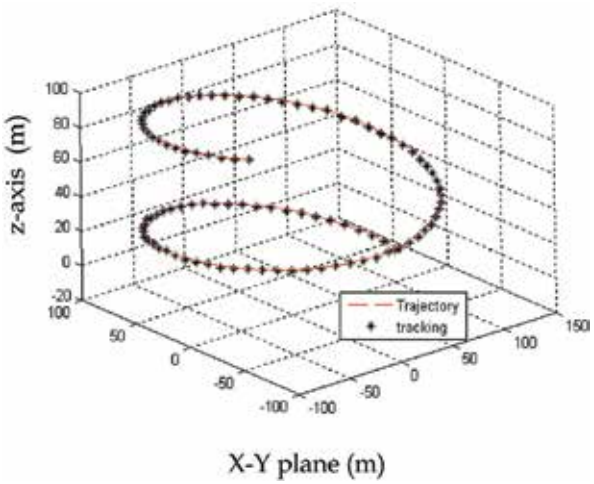


Fig. 6. (a) Trajectory-tracking of AUV moving in a spiral cord

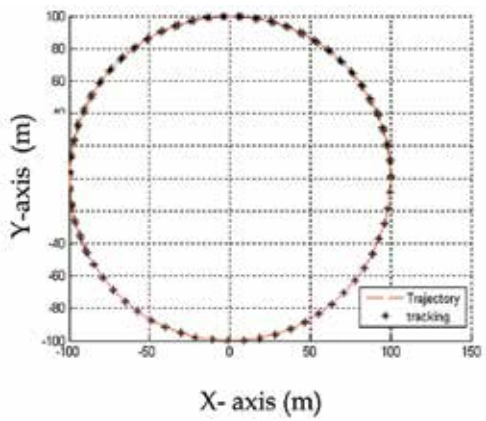


Fig. 6. (b) Trajectory-tracking of AUV projected in X-Y plane

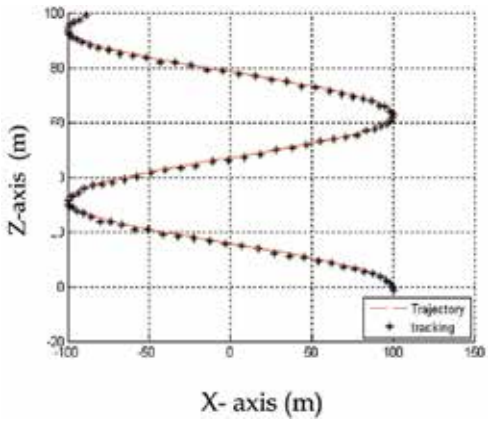


Fig. 6. (c) Trajectory-tracking of AUV projected in X-Z plane

6. Conclusion and future work

A new cooperative acoustic navigation concept based on intelligent sonobuoy with DGPS, that fits into coordinated control of hetergenous vehicles was proposed in this paper, which gives a great advantage to coordinate a team of heterogenous vehicles with provided position estimates in centi-metric accuracy. It is believed that the navigation method proposed may represent a significant alternative to the ones based on traditional navigation methods (such as LBL, SBL, USBL and INS/GPS, etc.), and the novel acoustic navigation method is especially suitable for the field of coordinated control of heterogenous vehicles including ASV and AUVs due to its characteristics of upwards acoustic flow of data.
Future work is to develop a robust acoustic navigation system that can perform reliably and also include a version of the system in the presence of multi-path effects, acoustic outliers and of course include in-water validation of the system.

7. Acknowledgement

The first author would like to acknowledge helpful discussion and solid support from the co-authors of the chapter.
The European Marie Curie Research Fellowship and the wonderful team in the research project FREESUBNET under grant 036186 give the first author a great support for the finalization of the chapter. This work was also funded in part by the National Natural Science Foundation of China under grant 69585003, and in part by the National 863 Hi-tech Research and Development Program of China under grant 2006AA09Z203.

8. References

Lapierre L. (2006). Underwater Robots Part I: current systems and problem pose, in *Mobile Robotics – Towards New Applications*, Edited by Pro Verlag & Advanced Robotic Systems International (ARS), ISBN: 978-3-86611-314-5, Germany.

Louis L. Whitcomb.(2000). Underwater Robotics: Out of the Research Laboratory and Into the Field, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 709-716, April 2000, San Francisco, USA.

Xu, G., Guo, Y., Xiang, X. and Xiao, Z (2007). Motion control and computer simulation for underwater vehicle-manipulator systems, *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, pp. 1368-1373. Harbin, China.

http://www.freesubnet.eu.

Xiang, X., Huang, X., Zhang, Q., Xu, G. (2008). Prospective research on coordinated control of multiple AUVS. *Proceedings of the 2008 IEEE Chinese Control and Decision Conference, pp. 3306-3310. Yantai, Shangdong, China.*

Bourgeois,B., Martinez,A., Alleman,P., Cheramie,J., Gravley,J. (1999). Autonomous bathymetry survey system. *IEEE Journal of Oceanic Engineering,* vol.24, no. 4, pp. 414-423.

McDowell,P., Chen,J., and Bourgeois,B. (2002). UUV Teams, control from a biological perspective. *Proceedings of the Oceans 2002 MTS/IEEE Conference, Biloxi MS,* pp 331-337.

Caiti,A., Garulli, A., Livide,F., and Prattichizzo,D. (2005). Localization of autonomous underwater vehicles by floating acoustic buoys: a set-membership approach. *IEEE Journal of Oceanic Engineering,* vol.30, no. 1, pp. 140-152.

Yun, X., Bachmann, E. R., McGhee, R. B., Whalen, R. H., Roberts, R. L., Knapp, R. G..,Healey, A. J., and Zyda, M. J. (1999). Testing and evaluation of an integrated gps/ins system for small auv navigation. *IEEE Journal of Ocean Engineering.,* vol. 24, no. 3, pp. 396-404.

Xiang, X., Xu, G., Zhang, Q., Guo, Y. and Huang, X. (2007). A novel acoustic navigation scheme for coordinated heterogenous autonomous vehicles. *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, pp. 2801-2806. Harbin, China.

Vickery,K. (1998). Acoustic positioning systems. A practical overview of current systems. *Proceedings of the 1998 Workshop on Autonomous Underwater Vehicles., Fort Lauderdale, FL, USA.*, pp. 5-17.

Collin, L., Azou, S., Yao, K., and Burel, G.. (2000). On spatial uncertainty in a surface long base-line positioning system. *Proc. 5th Europ. Conf. Underwater Acoustics*, Lyon, France.

Storkensen, N., Kristensen, J.,Indreeide, A., Seim, J. and Glancy, T. (1998). Hugin—UUV for seabed survey. *Journal of Sea Technology.*

Thomas, H. G.. (1998). GIB buoys: An interface between space and depths of the oceans. *Proceedings of IEEE Autonomous Underwater Vehicles, Cambridge, MA, USA*, pp.181-184.

http://www.underwater-gps.com.

Alcocer,A., Oliveira,P., and Pascoal,A. (2006). Underwater acoustic positioning system based on buoys with GPS. *Proc. 8th European Conf. on Underwater Acoustics,* Carvoeiro, Portugal.

Vaganay, J., Leonard, J., and Bellingham, J. (1996). Outlier rejection for autonomous acoustic navigation. *Proceedings of IEEE International Conference on Robotics and Automation,Minneapolis, MN, USA*, pp. 2174–2181.

Alcocer, A., Oliveira P., and Pascoal A. (2004). Study and implementation of an EKF GIB-Based underwater positioning system. *Proceedings of the IFAC Conference on Control Applications in Marine Systems (CAMS'04)*, Ancona, Italy.

Freitag,L., Johnson, M., Grund,M., Singh, S. and Preisig,J. (2001). Integrated acoustic communication and navigation for multiple UUVs, *Proc. Oceans 2001,* pp. 2065-2070, Honolulu, HI.

Foy,W.H.. (1976). Position-location solutions by Taylor-series estimation. *IEEE Trans. Aerospace and Electronic Systems*, vol.12, no.2, pp.187-194.

Xiang[a], X., Xu, G., Zhang, Q., Xiao, Z. and Huang, X. (2007). Coordinated control for multi-AUV systems based on hybrid automata. *Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics*, pp. 2121-2126. Sanya, China.

28

Dynamic Modelling and Motion Control for Underwater Vehicles with Fins

Xiao Liang, Yongjie Pang, Lei Wan and Bo Wang
Harbin Engineering University
China

1. Introduction

With the development of the activities in deep sea, the application of the autonomous underwater vehicle (AUV) is very widespread and there is a prominent prospect. The development of an AUV includes many areas, such as vehicle (carrier/platform) design, architecture, motion control, intelligent planning and decision making, etc (Blidberg 1991; Xu et al., 2006). The researchers dedicate themselves to improving the performance of modular, low-cost AUVs in such applications as long-range oceanographic survey, autonomous docking, and shallow-water mine countermeasures. These goals can be achieved through the improvement of maneuvering precision and motion control capability with energy constraints. For low energy consumption, low resistance, and excellent maneuverability, fins are usually utilized to modify the AUV hydrodynamic force. An AUV with fins can do gyratory motion by vertical fins and do diving and rising motion by horizontal fins. Therefore, the control system of the propeller-fin-drived AUV is very different to the conventional only-propeller-drived AUV.

A dynamic mathematic model for the AUV with fins based on a combination of theory and empirical data would provide an efficient platform for control system development, and an alternative to the typical trial-and-error method of control system tuning. Although some modeling and simulation methods have been proposed and applied (Conte et al., 1996; Timothy, 2001; Chang et al., 2002; Ridley, 2003; Li et al., 2005; Nahon, 2006; Silva et al., 2007), there is no standard procedure for modeling AUVs with fins in industry. Therefore, the simulation of the AUVs with fins is a challenge.

This chapter describes the development and verification of a six Degree of Freedom (DOF), non-linear model for an AUV with fins. In the model, the external force and moment resulting from hydrostatics, hydrodynamic lift and drag, added mass, and the thrusters and fins are all analyzed and expressed in matrix form. The equations describing the rigid-body dynamics are left in non-linear form to better simulate the AUV inherently non-linear behavior. Motion simulation is achieved through numeric integration of the motion equations. The simulation output is then checked with the AUV dynamics data collected in experiments at sea. The comparison results show that the non-linear model gives an accurate estimation of the AUV's acutal motion. The research objective of this project is the development of WEILONG mini-AUV, which is a small, low-cost platform serving in a range of oceanographic applications (Su et al., 2007).

Due to the effect of fins, conventional control methods can not meet the requirement for motion control (Giusepp, 1999). It requires high response speed and robustness to improve the maneuverability, at the same time the controller's compute process should be simple enough. This chapter proposes a new control method which is adaptive to the AUV with fins—S surface control (Liu et al., 2001). S surface controller is developed from sigmoid function and the idea of fuzzy control which has been proved efficient in ocean experiments. It has a simple structure requiring only two inputs, but it is applicable to nonlinear system. Moreover, we will deduce self-learning algorithm using BP algorithm of neural networks for reference (Liu et al., 2002). Finally, experiments are conducted on WEILONG AUV to verify the feasibility and superiority.

2. Mathematic modelling of AUV motion

2.1 Coordinate system and motion parameters definition

In order to describe the AUV motion and set up a 6-DOF nonlinear mathematical model, a special reference frames have been established (Shi, 1995). There are two reference frames: fixed reference frame E-$\xi\eta\zeta$ (or inertial coordinate system) and motion reference frame o-xyz (or body-fixed coordinate system), which are shown in Fig.1.

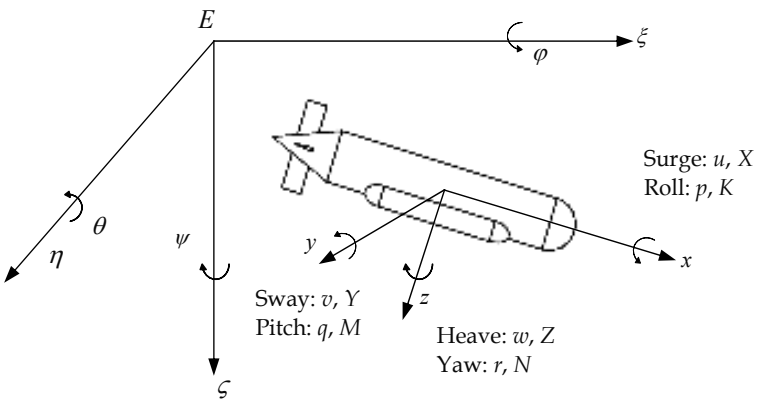


Fig. 1. Body-fixed and inertial coordinate system

Considering the shape characteristic of most AUVs, the mathematic model is based on the hypothesis that the AUV is symmetric about its xoz plane.

Defining generalized position vector $\mathbf{R}$, generalized velocity vector $\mathbf{V}$ and generalized force vector $\boldsymbol{\tau}$, the motion vector include

1. Position and attitude (in E-$\xi\eta\zeta$)

$$\mathbf{R} = [\mathbf{r}^T, \boldsymbol{\Lambda}^T]^T, \mathbf{r} = [\xi, \eta, \zeta]^T, \ \boldsymbol{\Lambda} = [\varphi, \theta, \psi]^T$$

2. Linear and angular velocities (in o-xyz)

$$\mathbf{V} = [\mathbf{U}^T, \boldsymbol{\Omega}^T]^T, \mathbf{U} = [u, v, w]^T, \boldsymbol{\Omega} = [p, q, r]^T$$

3. Force and moment parameters (in o-xyz)

$$\boldsymbol{\tau} = [\mathbf{F}^{\mathrm{T}}, \mathbf{M}^{\mathrm{T}}]^{\mathrm{T}}, \ \mathbf{F} = [X, Y, Z]^T, \ \mathbf{M} = [K, M, N]^T$$

2.2 Dynamics model

Based on momentum theorem, the AUV dynamic equation is

$$\mathbf{M}_{RB}\dot{\mathbf{V}} + \mathbf{C}_{RB}(\mathbf{V})\mathbf{V} = \boldsymbol{\tau} \tag{1}$$

where $\mathbf{M}_{RB}$ is the generalized mass matrix of the AUV body, and $\mathbf{C}_{RB}(\mathbf{V})$ is the Coriolis and centripetal force matrix. $\mathbf{M}_{RB}$ is given by

$$\mathbf{M}_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & J_x & J_{xy} & J_{xz} \\ mz_G & 0 & -mx_G & J_{yx} & J_y & J_{yz} \\ -my_G & mx_G & 0 & J_{zx} & J_{zy} & J_z \end{bmatrix} \tag{2}$$

where m is the AUV mass, J terms represent the inertial tensors, and x_G, y_G, z_G represent the AUV position barycenter in body-fixed frame. $\mathbf{C}_{RB}(\mathbf{V})$ is given by

$$\mathbf{C}_{RB}(\mathbf{V}) = \begin{bmatrix} 0 & -mr & mq \\ mr & 0 & -mp \\ -mq & mp & 0 \\ -m(y_G q + z_G r) & my_G p & mz_G p \\ mx_G u & -m(z_G r + x_G p) & mz_G q \\ mx_G r & my_G r & -m(x_G p + y_G q) \end{bmatrix} \begin{bmatrix} m(y_G q + z_G r) & -mx_G q & -mx_G r \\ -my_G p & m(z_G r + x_G p) & -my_G r \\ -mz_G p & -mz_G q & m(x_G p + y_G q) \\ 0 & J_{zx}p + J_{zy}q + J_z r & -J_{yx}p - J_y q - J_{yz}r \\ -J_{zx}p - J_{zy}q - J_z r & 0 & J_x p + J_{xy}q + J_{xz}r \\ J_{yx}p + J_y q + J_{yz}r & -J_x p - J_{xy}q - J_{xz}r & 0 \end{bmatrix} \tag{3}$$

Generalized force vector $\boldsymbol{\tau}$ at the right of the equation (1) is outside force (or moment) acting on the AUV, including static force vector $\boldsymbol{\tau}_G$ (gravity and buoyancy), hydrodynamics force vector of the vehicle body (include $\boldsymbol{\tau}_A$ which is caused by added mass and viscous damping force $\boldsymbol{\tau}_V$), and the controlled forece vector (include the thruster force $\boldsymbol{\tau}_{prop}$ and the fin force $\boldsymbol{\tau}_R$). The static force vector $\boldsymbol{\tau}_G$ reflects the effect of the vehicle weight and buoyancy. The vehicle's weight is $W = mg$ and the buoyancy is $B = \rho \nabla g$, where ρ is the dentisity of the surrounding fluid, and ∇ is the total volume displaced by the AUV. Therefore, $\boldsymbol{\tau}_G$ is given by

$$\boldsymbol{\tau}_G = \begin{bmatrix} X_G \\ Y_G \\ Z_G \\ K_G \\ M_G \\ N_G \end{bmatrix} = \begin{bmatrix} -(W - B)\cdot\sin\theta \\ (W - B)\cdot\sin\varphi\cos\theta \\ (W - B)\cdot\cos\varphi\cos\theta \\ (y_G W - y_B B)\cdot\cos\varphi\cos\theta - (z_G W - z_B B)\cdot\sin\varphi\cos\theta \\ -(x_G W - x_B B)\cdot\cos\varphi\cos\theta - (z_G W - z_B B)\cdot\sin\theta \\ (x_G W - x_B B)\cdot\sin\varphi\cos\theta - (y_G W - y_B B)\cdot\sin\theta \end{bmatrix} \tag{4}$$

where x_B, y_B, z_B are the vehicle coordinates in body-fixed coordinate system.
$\boldsymbol{\tau}_{\mathbf{A}}$ which is related with added mass is given by

$$\boldsymbol{\tau}_A = -(\mathbf{M}_A\dot{\mathbf{V}} + \mathbf{C}_A(\mathbf{V})\mathbf{V}) \tag{5}$$

where $\mathbf{M}_A$ is the added mass matrix given by

$$\mathbf{M}_A = \begin{bmatrix} \lambda_{11} & 0 & \lambda_{13} & 0 & \lambda_{15} & 0 \\ 0 & \lambda_{22} & 0 & \lambda_{24} & 0 & \lambda_{26} \\ \lambda_{31} & 0 & \lambda_{33} & 0 & \lambda_{35} & 0 \\ 0 & \lambda_{42} & 0 & \lambda_{44} & 0 & \lambda_{46} \\ \lambda_{51} & 0 & \lambda_{53} & 0 & \lambda_{55} & 0 \\ 0 & \lambda_{62} & 0 & \lambda_{64} & 0 & \lambda_{66} \end{bmatrix} \tag{6}$$

where λ terms are the vehicle added mass.
$\mathbf{M}_A$ can be also denoted as hydrodynamic coefficients expression as follows:

$$\mathbf{M}_A = -\begin{bmatrix} X_{\dot{u}} & 0 & X_{\dot{w}} & 0 & X_{\dot{q}} & 0 \\ 0 & Y_{\dot{v}} & 0 & Y_{\dot{p}} & 0 & Y_{\dot{r}} \\ Z_{\dot{u}} & 0 & Z_{\dot{w}} & 0 & Z_{\dot{q}} & 0 \\ 0 & K_{\dot{v}} & 0 & K_{\dot{p}} & 0 & K_{\dot{r}} \\ M_{\dot{u}} & 0 & M_{\dot{w}} & 0 & M_{\dot{q}} & 0 \\ 0 & N_{\dot{v}} & 0 & N_{\dot{p}} & 0 & N_{\dot{r}} \end{bmatrix} \tag{7}$$

$\mathbf{C}_A(\mathbf{V})$ is a Coriolis-like matrix induced by $\mathbf{M}_A$,

$$\mathbf{C}_A(\mathbf{V}) = \begin{bmatrix} 0 & 0 & 0 & 0 & a_3 & -a_2 \\ 0 & 0 & 0 & -a_3 & 0 & a_1 \\ 0 & 0 & 0 & a_2 & -a_1 & 0 \\ 0 & a_3 & -a_2 & 0 & b_3 & -b_2 \\ -a_3 & 0 & a_1 & -b_3 & 0 & b_1 \\ a_2 & -a_1 & 0 & b_2 & -b_1 & 0 \end{bmatrix} \tag{8}$$

where

$$a_1 = \lambda_{11}u + \lambda_{13}w + \lambda_{15}q \qquad a_2 = \lambda_{22}v + \lambda_{24}p + \lambda_{26}r \qquad a_3 = \lambda_{31}u + \lambda_{33}w + \lambda_{35}q$$

$$b_1 = \lambda_{42}v + \lambda_{44}p + \lambda_{46}r \qquad b_2 = \lambda_{51}u + \lambda_{53}w + \lambda_{55}q \qquad b_3 = \lambda_{62}v + \lambda_{64}p + \lambda_{66}r$$

The viscous damping force $\boldsymbol{\tau}_V$ is given by

$$\boldsymbol{\tau}_V = \mathbf{D}(\mathbf{V})\mathbf{V} \tag{9}$$

The damping matrix $\mathbf{D}(\mathbf{V})$ is given by

$$\mathbf{D}(\mathbf{V}) = \begin{bmatrix} X_u + X_{u|u|}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v + Y_{v|v|}|v| & 0 & 0 & 0 & 0 \\ Z_0|u| & 0 & Z_w + Z_{w|w|}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p + K_{p|p|}|p| & 0 & 0 \\ M_0|u| & 0 & 0 & 0 & M_q + M_{q|q|}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_r + N_{r|r|}|r| \end{bmatrix} \quad (10)$$

where X_u, Y_v, Z_w, K_p, M_q, and N_r are the linear damping coefficients. $Y_{v|v|}$, $Z_{w|w|}$, $K_{p|p|}$, $M_{q|q|}$, $X_{u|u|}$, and $N_{r|r|}$ are the quadratic damping coefficients. M_0 and Z_0 are the effect caused by the dissymmetry on *xoy* plane.

The external force and moment vector produced by trusters $\boldsymbol{\tau}_{prop}$ is defined as

$$\boldsymbol{\tau}_{prop} = \mathbf{L}\mathbf{T}_{prop} \quad (11)$$

where **L** is a mapping matrix, and $\mathbf{T}_{prop}$ is the thrust vector produced by thrusters given by

$$\mathbf{T}_{prop} = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} \quad (12)$$

The number *n* in $\mathbf{T}_{prop}$ depends on the number of thrusters. The mapping matrix **L** is a 6×n matrix that uses $\mathbf{T}_{prop}$ to find the overall force and moment acting on the vehicle.

Hydrodynamics of a single thruster is usually obtained through the in water test. A series of advance coefficient J corresponding to the thrust coefficient K_T data can be obtained from the in water test. Data from an in water test are shown in Fig.2. We fit the curve by the method of least squares and then obtain the fitted $J - K_T$ curve. In practical appilcations, we get advance coefficient J and substitute it into fitted $J - K_T$ curve to obtain K_T. Finally, the thrust can be obtained. Detailed process is as follows:

1. We get the advance coefficient J from fluid velocity cross the propeller V_{prop}, the propeller diameter D, and the screw propeller rotate speed n (n is determined by controller): $J = \dfrac{V_{prop}}{nD}$.
2. We put J into fitted $J - K_T$ curve to get force coefficient K_T.
3. We get thrust-**T** by using equation $\mathbf{T} = K_T n^2 D^4$.

The overall external force and moment vector produced by fins $\boldsymbol{\tau}_R$ is given by

$$\boldsymbol{\tau}_R = \begin{bmatrix} X_R \\ Y_R \\ Z_R \\ K_R \\ M_R \\ N_R \end{bmatrix} \quad (13)$$

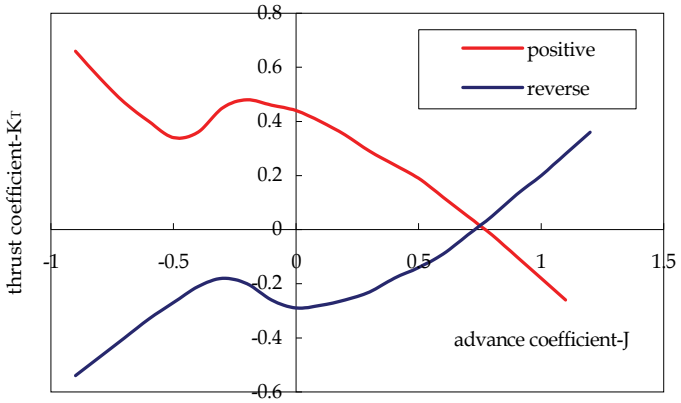


Fig. 2. Capability curves of thrusters

According to every single fin force and its installating position, $\boldsymbol{\tau}_R$ can be obtained.
As to a control fin on the vehicle, the hydrodynamic force can be decomposed into two directions: lift force **L**—vertical to stream current and drag force **D**—along stream current. Lift force and drag force can be calculated by the equations as follows:

$$\begin{aligned} \mathbf{L} &= \frac{1}{2} C_L \rho A_R v_e^2 \\ \mathbf{D} &= \frac{1}{2} C_D \rho A_R v_e^2 \end{aligned} \tag{14}$$

where C_L is the fin lift coefficient, C_D is the fin drag coefficient, A_R is the fin planform area, and v_e is the effective fin velocity. The values of lift coefficient C_L and drag coefficient C_D are related with effective fin angle of attack a.

We can adopt experiment, theroy computation, or empirical fomular to get C_L and C_D. Experiment and empirical fomular method will be introduced below.

1. Actual measurement from exprement

A series of data of angles of attack a vs. lift coefficient C_L and drag coefficient C_D can be obtained from hydrodynamic experiment, and then fitted curves of C_L and C_D can be generated through least squares fit. For example, the fitted currves of a fin is shown in Fig.3. When we know the current angle of attack of fin on the AUV, the values of C_L and C_D under this angle can be obtained by curves interpolation.

2. Method of empirical equation

The empirical equations to calculate C_L and C_D are given by

$$C_L = \frac{\partial C_L}{\partial \alpha} \times \alpha + \frac{C_{DC}}{\lambda} \left(\frac{\alpha}{57.3}\right)^2 \tag{15}$$

$$\frac{\partial C_L}{\partial \alpha} = \frac{0.9(2\pi)\lambda}{57.3\left[\cos\Lambda \sqrt{\frac{\lambda^2}{\cos^4\Lambda} + 4} + 1.8\right]} \tag{16}$$

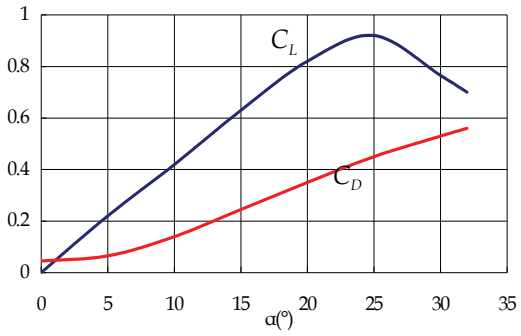


Fig. 3. Lift and drag coefficient curves

$$C_D = C_{d0} + \frac{C_L^{\,2}}{e\pi\lambda} \tag{17}$$

where $\frac{\partial C_L}{\partial \alpha}$ is the slope at a=0 in lift coefficient curves, and C_{DC} is the drag coefficient of cross current which depends on tip shape and rake ratio (e.g. Quadrate tip: C_{DC} =0.8. Smooth tip: C_{DC} =0.4). C_{d0} is the airfoil profile drag coefficient (viscous drag). For the profile section NACA0015, C_{d0} =0.0065. Λ is the sweptback angle at 1/4 chord of the fin. λ is the aspect ratio. a is the angle of attack (degree).

In order to get C_L and C_D, we should know the real effective angle of atttack. As the fin located at some offset from the origin of the AUV coordinate system, it experiences the following effective velocities

$$\begin{aligned} u_{fin} &= u + z_{fin} - y_{fin} r \\ v_{fin} &= v + x_{fin} r - z_{fin} p \\ w_{fin} &= w + y_{fin} - x_{fin} q \end{aligned} \tag{18}$$

where x_{fin}, y_{fin}, and z_{fin} are the body-fixed coordinates of the fin posts.

The effective fin angles of attack δ_{se} and δ_{re} are given by

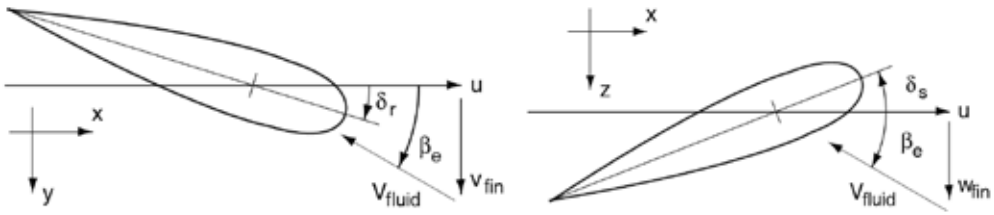
$$\begin{aligned} \delta_{se} &= \delta_s + \beta_{se} \\ \delta_{re} &= \delta_r + \beta_{re} \end{aligned} \tag{19}$$

where δ_r and δ_s are the fin angles referenced to the vehicle hull, β_{re} and β_{se} are the effective angles of attack of the fin zero plane, as shown in Fig.4. β_{re} and β_{se} are given by

$$\beta_{re} = \frac{v_{fin}}{u_{fin}} = \frac{v + x_{fin} r - z_{fin} p}{u + z_{fin} - y_{fin} r} \qquad \beta_{se} = \frac{w_{fin}}{u_{fin}} = \frac{w + y_{fin} - x_{fin} q}{u + z_{fin} - y_{fin} r} \tag{20}$$

Based on the above analysis, equation (1) could be rewritten into more detailed form

$$(\mathbf{M}_{RB} + \mathbf{M}_A)\dot{\mathbf{V}} = \boldsymbol{\tau}_G + \boldsymbol{\tau}_{prop} + \boldsymbol{\tau}_R + \mathbf{D}(\mathbf{V})\mathbf{V} - (\mathbf{C}_{RB}(\mathbf{V}) + \mathbf{C}_A(\mathbf{V}))\mathbf{V} \tag{21}$$



(a) Effective rudder angle of attack (b) Effective stern plane angle of attack

Fig. 4. Effective angle of attack scheme

2.3 Kinematics model

The coordinate transformation between body-fixed coordinate system and inertial coordinate system can is given by

$$\begin{bmatrix} \dot{\xi}_G \\ \dot{\eta}_G \\ \dot{\zeta}_G \\ \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} T_1 & 0_{3\times3} \\ 0_{3\times3} & T_2 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \tag{22}$$

where ξ_G, η_G and ζ_G are the barycentre coordinates in inertial coordinate system, T_1 and T_2 are coordinate transform matrix given by

$$\boldsymbol{T}_1 = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi & \cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\varphi + \cos\psi\cos\varphi & \sin\psi\sin\theta\cos\varphi - \cos\psi\sin\varphi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \tag{23}$$

$$\boldsymbol{T}_2 = \begin{bmatrix} 1 & \tan\theta\sin\varphi & \tan\theta\cos\varphi \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi\sec\theta & \cos\varphi\sec\theta \end{bmatrix} \tag{24}$$

2.4 Numerical integration

Given the complex and highly nonlinear nature of the equations (21) and (22), we will use numerical integration to solve these equations and get the vehicle speed, position, and attitude vs time.

The non-linear state equation of the AUV is given by

$$\dot{\mathbf{x}}_n = \mathrm{f}(\mathbf{x}_n, \mathbf{u}_n) \tag{25}$$

where $\mathbf{x}_n$ is the state vector, and $\mathbf{u}_n$ is the input vector:

$$\mathbf{x}_n = [u \quad v \quad w \quad p \quad q \quad r \quad \xi \quad \eta \quad \zeta \quad \varphi \quad \theta \quad \psi]^T \tag{26}$$

$$\mathbf{u}_n = [\boldsymbol{\tau}_{prop} \quad \boldsymbol{\tau}_R] \tag{27}$$

Here, Runge-Kutta method of numerical integration is usually used to solve the equations. Firstly, we calculate the following equations

$$\begin{aligned} k_1 &= \boldsymbol{x}_n + \mathrm{f}(\mathbf{x}_n, \mathbf{u}_n) \\ k_2 &= \mathrm{f}(\mathbf{x} + \frac{\Delta t}{2} k_1, \mathbf{u}_{n+\frac{1}{2}}) \\ k_3 &= \mathrm{f}(\mathbf{x} + \frac{\Delta t}{2} k_2, \mathbf{u}_{n+\frac{1}{2}}) \\ k_4 &= \mathrm{f}(\mathbf{x} + \Delta t k_4, \mathbf{u}_{n+1}) \end{aligned} \tag{28}$$

where the interpolated input vector is

$$\mathbf{u}_{n+\frac{1}{2}} = \frac{1}{2}(\mathbf{u}_n + \mathbf{u}_{n+1}) \tag{29}$$

Then, we combine the above equations

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{30}$$

2.5 Simulation results

Base on the above mathematic modelling and analysis, many simulation data are obtained using the simulator of one AUV. The simulation results are compared with the results of at-sea experiments. The zigzag-like motions in horizontal plane and vertical plane were simulated and the compared results are shown in Fig.5 and Fig.6.

From the comparison between simulation results and experiment results, we can conclude that the mathematic model of the AUV motion and the numerical integration method are accurate and feasible.

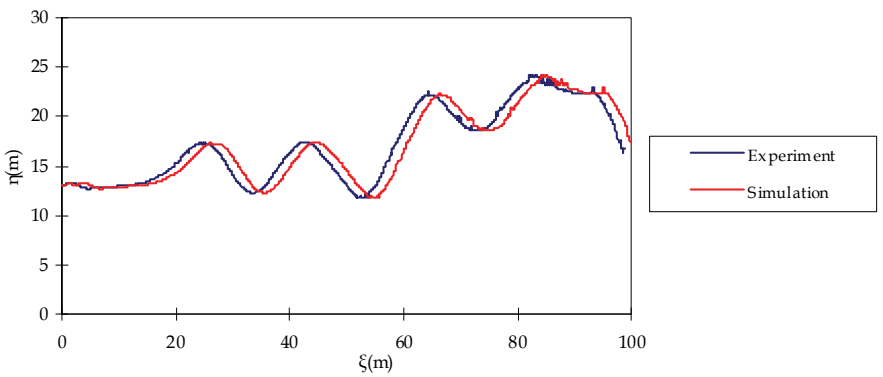


Fig. 5. Zigzag-like motion in horizontal plane

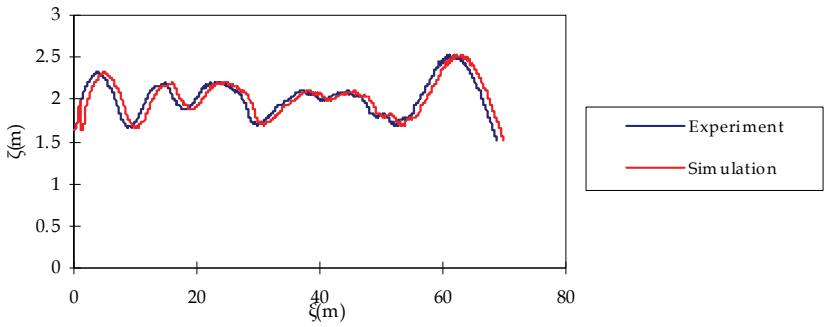


Fig. 6. Zigzag-like motion in vertical plane

2.6 Summary

The nonlinear mathematic model of the AUV motion is constructed in this section and the numerical integration to solve this model is also discussed. Moreover, the model is applicable to most AUVs.

3. AUV motion control

3.1 S surface control

The control rules (Table 1) of the general fuzzy controller indicate that changes of the control outputs are regular. Based on the figures along the leading diagonal, there is a polygonal line, which can be fitted with a smooth curve (a sigmoid function). In fact, the smooth curve can be viewed as innumerable polygonal lines with a length approaching to zero joined together. When designing fuzzy controller, the form (when the deviation is comparatively large, the control demand would be loosely considered; on the contrary, when the deviation is comparatively small, the control demand would be strictly treated) that is loosen at both sides and thick at the middle is generally adopted, which is consistent with the variation form of the sigmoid function. Thus, the sigmoid function incarnates the idea of fuzzy control on a certain extent. Moreover, the fold line surface that corresponds with the whole fuzzy rule of fuzzy control can be replaced with the curved surface composed by smooth curves, as shown in figure 7.

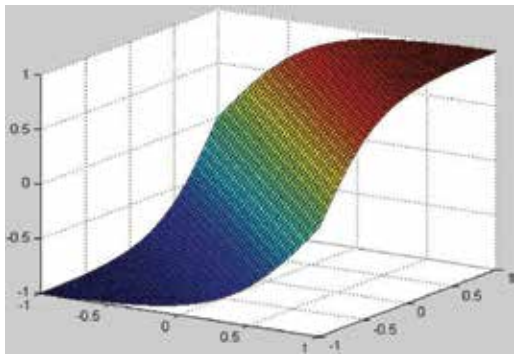


Fig. 7. Sigmoid curved surface

4	3	2	1
3	2	1	0
2	1	0	-2
1	0	-1	-2
0	-1	-2	-3

Table 1. Control rules table

Generally, the function of sigmoid curve is given by

$$y = 2.0/\left(1.0 + e^{-kx}\right) - 1.0 \tag{31}$$

Then, the function of sigmoid curved surface is

$$z = 2.0/\left(1.0 + e^{(-k_1 x - k_2 y)}\right) - 1.0 \tag{32}$$

Thus, the designed control model of S surface controller is

$$u = 2.0/\left(1.0 + e^{(-k_1 e - k_2 \dot{e})}\right) - 1.0 \tag{33}$$

where e and $\dot{e}$ stand for the input information (error and the rate of error change, which are normalized), u is the control output which is the output force (normalized) in each freedom, and k_1 and k_2 are the control parameters corresponding to error and rate of error change respectively.

In equation (33), there are only two control parameters (k_1 and k_2) which S surface controller need to adjust. It is important to note that S surface controller can not get the best matching, whether adopting manual adjustment or adaptive adjustment. This is because that the adjustment is global and local adjustment is not available. Therefore, parameter adjustment is just the approximation of the system. After all, due to the complexity and uncertainty of control object, any kind of approach has big approximation. Thus, the optimal parameters k_1 and k_2 are different due to different velocities.

Manual adjustment of control parameters can make the motion control of underwater vehicle meet the demand in most cases. Response is more sensitive to small deviation but vibrations easily occur when k_1 and k_2 are larger. Therefore, the initial values of k_1 and k_2 we choose are generally about 3.0. If the overshoot is large, we can reduce k_1 and increase k_2 simultaneously. By contrast, if the speed of convergence is slow, we can increase k_1 and reduce k_2 simultaneously.

The ocean current and unknown disturbances can be considered as fixed disturbance force in a samlping period. Thus, we can eliminate the fixed deviation by adjusting the excursion of S surface and the function of control model is

$$u = 2.0/\left(1.0 + e^{(-k_1 e - k_2 \dot{e})}\right) - 1.0 + \Delta u \tag{34}$$

where Δu is the value(normalized) of fixed disturbance force which is obtained through adaptive manner. The adaptive manner is as follows:

a. Check whether the velocity of the vehicle is smaller than a preset threshold. If it is, go to step b), if not, go to step c);
b. Give the deviation value of this degree to a set array, at the same time, add 1 to the set counter, when the very counter reaches the predefined value, go to step d);
c. Shift each element in the array to the left by one, and at the meantime, decrease the counter by 1, then go to step a);
d. Weighted average the values of the array and the gained average deviation values are obtained. Then these deviation values are used to compute the side-play amount of control output, self-adapt the control output to eliminate fixed deviation, meanwhile, set the counter to zero, turn to the next loop.

Thus, a simple and practical controller is constructed, which can meet the work requirement in complicated ocean environment. However, the parameter adjustment of S surface controller is completely by hand. We hope to adjust the parameters for the controller by itself online, so we will present the self-learning algorithm the idea borrowed from BP algorithm in neural networks.

3.2 Self-learning algorithm

Generally, we define a suitable error function using neural networks for reference, so we can adjust the control parameters by BP algorithm on-line. As is known, an AUV has its own motion will, which is very important for self-learning and will be discussed in detail in the next section, so there is also an expected motion state. Namely, there is an expected control output for S surface controller. Therefore, the error function is given by

$$E_p = \frac{1}{2}(u_d - u)^2 \tag{35}$$

where u_d is the expected control output, and u is the last time output which can be obtained by eqution (34) .

We can use gradient descent optimization method, i.e. use the gradient of E_p to adjust k_1 and k_2.

$$\Delta k_i = -\eta \frac{\partial E_p}{\partial k_i} \tag{36}$$

where η is the learning ratio ($0 < \eta < 1$).

$$\frac{\partial E_p}{\partial k_i} = -(u_d - u) \cdot \frac{\partial u}{\partial k_i} = -(u_d - u) \cdot \frac{2.0 e^{-k_1 e - k_2 \dot{e}}}{(1 + e^{-k_1 e - k_2 \dot{e}})^2} e_i \tag{37}$$

where $i = 1,2$; $e_1 = e$; $e_2 = \dot{e}$

Therefore, k_1 and k_2 can be optimized by the following eqution.

$$k_i(t+1) = k_i(t) + \Delta k_i = k_i(t) + \eta(u_d - u) \cdot \frac{2 e^{-k_1 e - k_2 \dot{e}}}{(1 + e^{-k_1 e - k_2 \dot{e}})^2} \cdot e_i \tag{38}$$

We can get the expected speed by expected state programming. The expected control output can be obtained by the following principles.

If the speed v is less than or equal to v_d, then u is less than u_d, and u needs to be magnified. In the contrast, u needs to be reduced. The expected control output is given by

$$u_d = u + c \cdot (v_d - v) \tag{39}$$

where c is a proper positive constant. Therefore, S surface controller has the ability of self-learning.

3.3 AUV motion will

As an intelligent system, the AUV has motion will to some degree. It knows the expected speed and when and how to run and stop. The effect from environment changing is secondary, and it can overcome the distubance by itself. Certainly, the obility to overcome the distubance is not given by researchers, because they may not have the detailed knowledge of the changing of environment. Howerver, the AUV motion will can be given easily, because the artificial machine must reflect the human ideas. For example, when an AUV runs from the current state to the objective state, how to get the expected acceleration(motion will) can be considered synthetically by the power of thrusters, the working requirement and the energy consumption. However, the active compensation to various acting force (the reflective intelligence for achieving the motion will) will be obtained from self-learning. This is the path which we should follow for the AUV motion control (Peng, 1995).

The purpose of motion control is to drive the error **S** and the error variance ratio **V** between the current state and and the objective state to be zero. The pre-programming of control output is given by

$$\mathbf{a} = \mathbf{V} = \{a_x, a_y, a_z, a_\psi, a_\theta\} = f(\mathbf{S}, \mathbf{V}) \tag{40}$$

where the concrete form of $f(\cdot)$ can be given by synthetically consideration according to the drive ability of the power system.

$$\mathbf{a} = \mathbf{P}\mathbf{a}_{\max} \tag{41}$$

where $\mathbf{a}_{\max}$ is the AUV maximal acceleration, which lies on the drive ability of power system and the vehicle mass. **P** is given by

$$\mathbf{P} = \begin{bmatrix} p_1 & & & & & 0 \\ & p_2 & & & & \\ & & p_3 & & & \\ & & & p_4 & & \\ 0 & & & & & p_5 \end{bmatrix} \tag{42}$$

where

$$\begin{cases} p_1 = (p_x / p_{xy}) \tanh(p_{xy} / 2) \\ p_2 = (p_y / p_{xy}) \tanh(p_{xy} / 2) \\ p_3 = \tanh(p_z / 2) \\ p_4 = \tanh(p_\psi / 2) \\ p_5 = \tanh(p_\theta / 2) \end{cases} \tag{43}$$

$$\begin{cases} p_x = S_x^* - c_x V_x \\ p_y = S_y^* - c_y V_y \\ p_z = S_z^* - c_z V_z \\ p_\psi = S_\psi^* - c_\psi V_\psi \\ p_\theta = S_\theta^* - c_\theta V_\theta \\ p_{xy} = \sqrt{p_x^2 + p_y^2} \end{cases} \tag{44}$$

where S_x^*, S_y^*, S_z^*, S_ψ^*, S_θ^* are difined as the traction distances in x, y, z, ψ, θ direction given by

$$S_i^* = \begin{cases} S_{i\max}^* & (S_i \geq S_{i\max}^*) \\ S_i & (-S_{i\max}^* < S_i < -S_{i\max}^*) \\ -S_{i\max}^* & (S_i \leq -S_{i\max}^*) \end{cases} \tag{45}$$

where $i = x$, y, z, ψ, θ. $S_{i\max}^*$ and c_i are undetermined coefficients, and $S_{i\max}^*$ are the predefined maximal distances which are determined based on the AUV's ability. We hope that the maximal transfer speed $V_{i\max}$

$$S_{i\max}^* - c_i V_{i\max} = 0 \tag{46}$$

As can be seen, we can not determine $S_{i\max}^*$ and c_i by equation (46), so we define the other constraint equation shown in equation (47).

$$\begin{cases} S_i'' = a_{i\max}\left(1 - \dfrac{2}{1+\exp\left(c_i S_i' - S_i\right)}\right), & (t > t_0) \\ S_i = S_{i\max}^* \\ S_i' = V_{i\max}, & (\mathrm{t} = \mathrm{t}_0) \end{cases} \tag{47}$$

Therefore, to all $t > t_0$, $S_i > 0$, and get smallest possible $t_n > t_0$. To all $t > t_n$, we can obtain

$$S_i < \varepsilon_i \tag{48}$$

where ε_i is the state precision. The constraint condition is to reduce errors as well as drive overshoot to zero.

4. Experiments

In this part, simulation and lake experiments have been conducted on WEILONG mini-AUV for many times to verify the feasibility and superiority of the mathmetical modelling and control method. The position errors of longitudinal control simulation are shown in Fig. 8. Reference inputs are 5m, the velocity of current is 0 m/s, and the voltage of thrusters is restricted by 2.5V. As can be seen, S surface control is feasible for the AUV motion control.
For the figure on the left, $k_1 = 8.0$ and $k_2 = 5.0$. Since the initial parameters are too big, there is certain overshoot and concussion aroud the object state in S surface control. However, the

parameters are adjusted by self-learning in improved S surface control. The overshoot is reduced and the balance (? Do you mean steady state) is achieved rapidly. For the figure on the right, $k_1 = 3.0$ and $k_2 = 5.0$. The initial parameters are too small, so the rate of convergence is too slow in S surface control. In improved S surface control, the rate of convergence is picked up and the performance is improved greatly.

Field experiments are conducted in the lake. The experiments use the impoved S surface control and the results are shown in Fig. 9 and Fig. 10. As there exits various disturbance (such as wave and current), the result curves are not smooth enough. In yaw control experiment, the action of the disturbances is greater than the acting force, so we can see some concussions in Fig. 9. It needs to be explained in the depth control that there is no response at the beginning of the experiment. The reason is the velocity of WEILONG mini-AUV is very low and the fin effect is too small. In the computer simulation, we don't use the fins until the velocity reaches certain value.

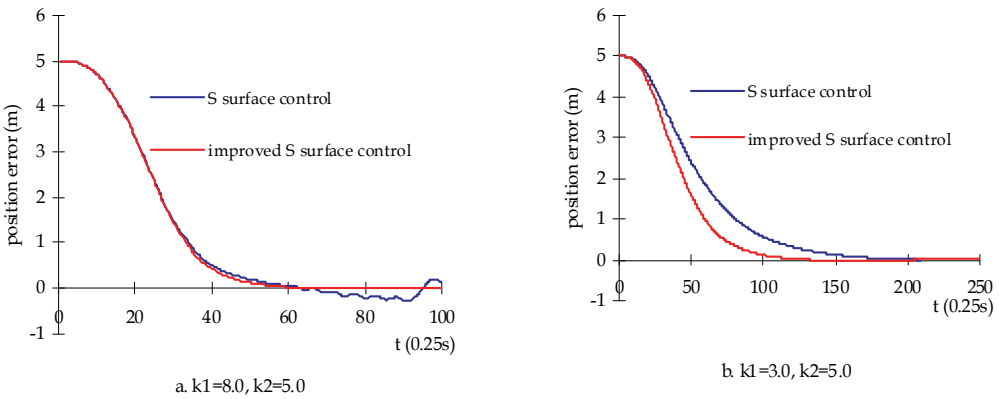


Fig. 8. Simulation results of longitudinal control

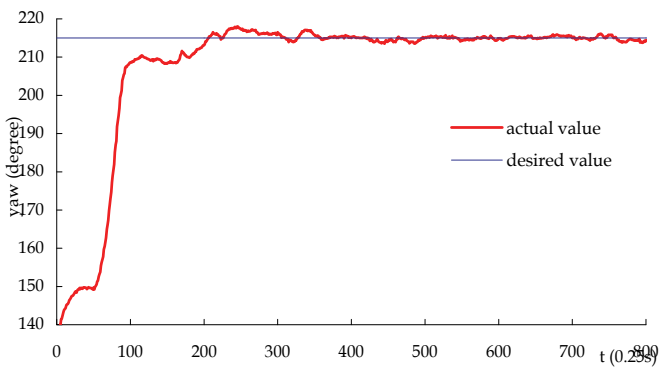


Fig. 9. Results of yaw control in lake experiments

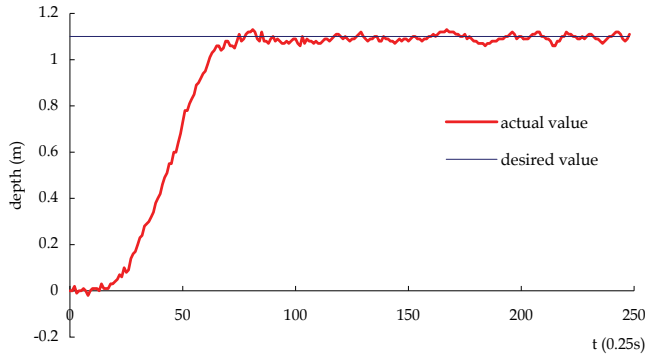


Fig. 10. Results of depth control in lake experiments

As can be seen, the control performance meets the requirement for the AUV motion control by using improved S surface control. It has high response speed and good robustness to various disturbances in field experiments.

5. Conclusion

This chapter concentrates on the problem of modeling and motion control for the AUVs with fins. Firstly, we develop the motion equation in six-degree freedom and analyze the force and hydrodynamic coefficients, especilly the fin effect. The feasibility and accuracy are verified by comparing the results between at-sea experiments and simulation. The model is applicable to most AUVs. Secondly, we present a simple and practical control method—S surface control to achieve motion control for the AUVs with fins, and deduce the self-learning algorithm using BP algorithm of neural networks for reference. Finally, the experiment results verify the feasibility and the superiority of the mathmetical modelling and control method.

6. Acknowledgements

The authors wish to thank all the researchers at the AUV Lab in Harbin Engineering University without whom it would have been impossible to write this chapter. Specifically, the authors would like to thank Professor Yuru Xu who is the subject leader of Naval Architecture and Ocean Engineering in Harbin Engineering University and has been elected as the member of Chinese Academy of Engineering since 2003. Moreover, the authors would like to thank Pang Shuo who is an assistant professor of Embry-Riddle Aeronautical University in USA.

7. References

Blidberg D.R. (1991). Autonomous underwater vehicles: a tool for the ocean, *Unmanned Systems*, Vol. 9, No. 2, 10-15, 1991.

Xu Y.R.; Pang Y.J.; Gan Y. & Sun Y.S. (2006). AUV-state-of-the-art and prospect. *CAAI Transactions on Intelligent Systems*, Vol.1, No.1, 9-16, September 2006.

Xu Y.R. & Xiao K. (2007). Technology development of autonomous ocean vehicle. *Journal of Automation*, Vol. 33, No. 5, 518-521, 2007.

Conte G. & Serrani A. (1996). Modelling and simulation of underwater vehicles. *Proceedings of the 1996 IEEE International Symposium on Computer-Aided Control System Design*, pp. 62-67, Dearborn, Michigan, September 1996

Timothy P. (2001). Development of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle: Oceans. *MTS/IEEE Conference and Exhibition*, pp. 450-455, May 2001

Prestero T. J. (2001). Development of a six-degree of freedom simulation model for the remus autonomous underwater vehicle. *Proceedings of the OCEANS 2001 MTS/IEEE Conference and Exhibition*, pp. 450-455, Honolulu, Hawaii, November 2001

Ridley P.; Fontan J. & Corke P. (2003). Submarine dynamic modeling. *Proceedings of the Australian Conference on Robotics and Automation*, Brisbane, Australia, December 2003

Chang W.J.; Liu J.C. & Yu H.N. (2002). Mathematic model of the AUV motion control and simulator. *Ship Engineering*, *y*, Vol.12, No.3, 58-60, September 2002.

Li Y.; Liu J.C. & Shen M.X.(2005). Dynamics model of underwater robot motion control in 6 degrees of freedom. *Journal of Harbin Institute of Technology*, Vol.12, No.4, 456-459, December 2005.

Nahon M. (2006). A Simplified Dynamics Model for Autonomous Underwater Vehicles. *Journal of Ocean Technology*, Vol. 1, No. 1, pp. 57-68, 2006

Silva J.; Terra B.; Martins R. & Sousa J. (2007). Modeling and Simulation of the LAUV Autonomous Underwater Vehicle. *Proceedings of the 13th IEEE IFAC International Conference on Methods and Models in Automation and Robotics*, pp. 713-718, Szczecin, Poland, August 2007

Su Y.M.; Wan L. & Li Y. (2007). Development of a small autonomous underwater vehicle controlled by thrusters and fins. *Robot*, Vol. 29, No. 2, 151-154, 2007.

Shi S.D. (1995). *Submarine Maneuverability*. National Defence Industry Press, Beijing.

Louis A.G. (2004). Design, modelling and control of an autonomous underwater vehicle. *Bachelor of engineering honours thesis*, University of Western Australia, 2004.

Giuseppe C. (1999). Robust Nonlinear Motion Control for AUVs. *IEEE Robotics & Automation Magazine*. pp. 33-38, May 1999

Peng L.; Lu Y.C. & Wan L. (1995). Neural network control of autonomous underwater vehicles. *Ocean Engineering*, Vol.12, No.2, 38-46, December 1995.

Liu X.M. & Xu Y.R. (2001). S control of automatic underwater vehicles. *Ocean Engineering*, Vol.19, No.3, 81-84, September 2001.

Liu J.C.; Yu H.N. & Xu Y.R. (2002). Improved S surface control algorithm for underwater vehicles. *Journal of Harbin Engineering University*, Vol.23, No.1, 33-36, March 2002.

29

Fundamentals of Underwater Vehicle Hardware and Their Applications

Hiroshi Yoshida
Japan Agency for Marine-Earth Science and Technology
Japan

1. Introduction

The evolution of electrical and electronic engineering technology including nanotechnology over the last several years has led to improvements in the development of mobile underwater platforms or autonomous underwater vehicles (AUVs) enabling them to go where tethered vehicles or manned vehicles have trouble reaching, such as under the ice, other dangerous zones, and into the deepest depths. In order to survey the whole ocean efficiently, the development of intelligent underwater vehicles will be one necessary solution. For the development of practical intelligent underwater vehicles, designers need cutting-edge fundamental devices incorporated into advanced underwater vehicles. Over the past ten years, the underwater research and development team to which the author belongs has developed five custom-made underwater vehicles: *Urashima* (Aoki 2001 & 2008), *UROV7k* (Murashma 2004), *MR-X1 (*Yoshida 2004), *PICASSO*, and *ABISMO*.
Urashima is the prototype vehicle of a long cruising range AUV (LCAUV) powered by the hybrid power source of a lithium-ion battery and a fuel cell. *Urashima* autonomously travelled over 300 km for about 60 hours in 2005. The LCAUV aims to make surveys under the arctic ice possible for distances of over 3000 km. The *UROV7k* is a tether cable-less ROV, having its power source in its body like an AUV. The *UROV7k* was designed to dive up to 7000 m without large on-board equipment such as a cable winch, a traction winch or a power generator. The *MR-X1* is a middle-size prototype AUV for the test of modern control methods and new hardware and for the development of new mission algorithms. The plankton survey system development project named Plankton Investigatory Collaborating Survey System Operon (*PICASSO*) project at the Japan Agency for Marine-earth Science and TEChnology (JAMSTEC) aims to establish a multiple vehicle observation system for efficient and innovative research on plankton. By using the ROV *Kaiko*, which was the deepest diving ROV in the world, a number of novel bacteria were found from mud samples taken in the Challenger Deep in the Mariana Trench (Takai, 1999). However, the lower vehicle of the *KAIKO* system was lost when the secondary tether was sheared (Watanabe 2004). The most important goal of the *ABISMO* system is to obtain mud samples from the Challenger Deep in the Mariana Trench, because scientists still want uninterrupted access to the deepest parts of the oceans using a vehicle equipped with sediment samplers. *ABISMO* consists of a sampling station and a sediment probe. The station contains two types of bottom samplers. One launches the probe to make a preliminary survey, launching the sampler to obtain a sample.

Through the development of these vehicles, many improvements in fundamental devices for underwater vehicles were made. In this chapter, firstly, hardware information on the key devices needed to make cutting edge intelligent underwater vehicles are described. These include new original devices: a small electrical-optical hybrid communication system, an HDTV optical communication system, an inertial navigation system, buoyancy material for the deepest depths, a thin cable with high-tensile strength, a USBL system, a broadcast class HDTV camera system, an HDTV stereoscopic system, a high capacity lithium ion battery, a high efficiency closed-cycle PEM fuel cell, and a prototype of an underwater electromagnetic communication system. In the third section, we present attempts made for data processing methods for autonomous control of underwater vehicles. Finally, the details of the AUVs using the above-mentioned devices are given, including some of the sea trial results.

2. Underwater vehicle hardware

2.1 Categories of unmanned underwater vehicles and their basic device components

Remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) are well-known kinds of underwater vehicles. Recently, there are also newer categories of underwater vehicles, untethered ROVs (UROVs) and hybrid ROVs (HROVs). UROVs (Aoki et al., 1992) have the feature that the vehicle is only connected to its support ship via a long thin optical fiber cable. The vehicle of an UROV system has its own power supply, in the form of batteries - much like an AUV. An operator controls the vehicle in real-time and has access to high quality real-time video images using high data rate optical communication tools. UROVs have both the advantages of ROVs and AUVs. An HROV (Bowen et al., 2004), one of which is under development at the Woods Hole Oceanographic Institution, is a single vehicle that can perform two different, but related, missions. It refers to the vehicle's ability to do scientific research while tethered to the ship, and also while swimming freely. Traditionally, a separate vehicle is used to conduct long range surveys, while another vehicle performs the close-up work and sampling. The HROV will simply transform between its two modes of operation to accomplish both of these tasks. In this section, cutting edge basic devices, except for those devices used for controlling vehicles and power sources, are described.

a. Buoyancy Materials and Cables

These are fundamental devices for underwater vehicles. In extreme environments, such as in the deepest depths, a developer should use special devices to match the mission. Full depth buoyancy materials have been commercialized but they have never actually been used in real situations at full ocean depth. The HROV project group at WHOI has chosen SeaSpheres, produced by Deepsea Power & Light, as an alternative to syntactic foams made from micro glass balloons. JAMSTEC has developed a new buoyancy material usable at full ocean depth. The prototype was used in the ABISMO system and it successfully withstood a 10,300 m depth deployment in 2008. The specifications of the prototype are a crush pressure of 56 MPa and a specific gravity of 0.63.

Tether cables for underwater vehicles are also a key device for successful development. Many companies have produced underwater cables, except for cables rated for full depth. Kyo (Kyo 1999) used a Kevlar fiber cable for the full depth vehicle *Kaiko*, but it was broken during retrieval of the *Kaiko* vehicle in the face of an approaching typhoon (Watanabe 2004). JAMSTEC thus started the development of a new cable using para-aramid fiber with a

tensile strength of 350kg/mm² in 2005. This rod type aramid fiber does not concentrate stress. The cable (ϕ20 mm x 160 m) consists of this aramid fiber, two coaxial cables, four single wire cables for power lines, cable sheath, and resin. The cable is covered in polypropylene. Specific gravity of the cable is around 1.3 and rupture strength is about 70 kN.

Fig. 1. A prototype of the full ocean depth buoyancy material (left) and the secondary cable made from para-aramid fiber (right).

Thin fiber optic cable and spoolers are used for UROV and HROV systems. Traditional ϕ0.9 mm single mode fiber (Murashima 2004) or thinner fiber cable (Young 2006) is practically used for underwater vehicles.

b. Lights and Cameras

For the observation of marine organisms, seafloor geology and underwater object recognition, the selection and arrangement of lights and cameras are important. The popularity of high definition television (HDTV) cameras and LED lights are causing an increase in availability of underwater video. In addition to high quality camera imaging, there are holographic cameras, laser scanning systems, acoustic imaging systems and so on. Further information on these imaging systems has been reviewed by Kocak et al. (2008).

The underwater vehicle *PICASSO*, developed by JAMSTEC (Yoshida 2007), is equipped with a broadcast quality HDTV camera. This high resolution, high sensitivity camera enables precise observation of plankton beyond that which was possible with traditional NTSC cameras. The increase in resolution means animals can be identified to species rather than genus or simply family in some cases. JAMSTEC has developed an original wideband optical communication system with five interfaces: one HD-SDI, three NTSCs, four RS-232Cs, two RS-485s, and 8-channel parallel I/O for the vehicle. This system will be discussed later. They installed SONY's compact high definition camera system, HDC-X300K, and an original camera control board with a CAN interface into an aluminum pressure hull. A special coaxial underwater cable with pressure-tight SMB type RF connectors was made for connecting between pressure hulls. HDC-X300 has the following specifications: effective pixels 1440×1080, sensitivity of 2000 lx @ F10, minimum luminance of 0.003 lx @ F1.4, smear level of -120 dB , and signal to noise ratio of 52 dB. Its image sensor system consists of three 1/2" 1.5M-pixel CCDs. Remote control of the focus, iris, and zoom of this camera via the original control board is possible. The HD-SDI output signal the camera is directly transmitted to an on-board system as an optical modulation signal via the optical communication system. The HD-SDI signal, demodulated and output from the on-board system, is connected to both of an HDCAM recorder and an HDTV display. Any movie subjects are lighted using HID lamps (three custom 30 watt lamps diverted from car use) and/or handmade 20 watts LED array lights. Examples of captured HDTV images obtained by *PICASSO* are shown in Figure 2.

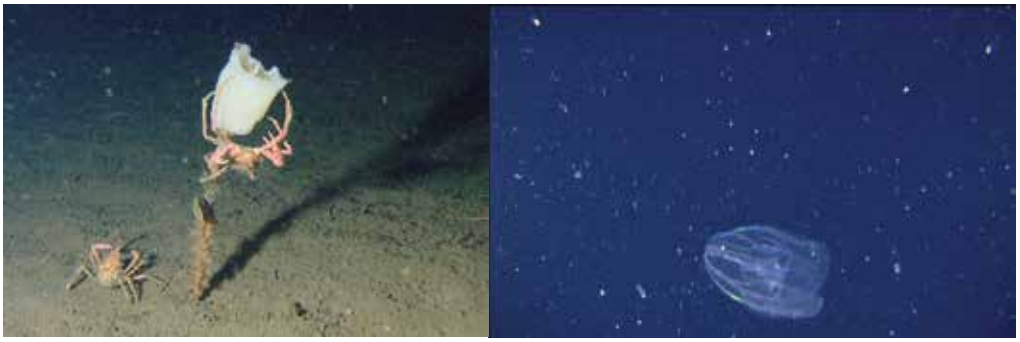


Fig. 2. An examples of an HDTV images taken by *PICASSO-1*. In this picture, the sponge and crabs are illuminated by a single HID lamp (left).

High power white LEDs, originally developed by Nichia corporation, have become widely used. Many underwater device makers produce underwater LED lights but they may be expensive. A low cost LED array in an oil-filled pressure balanced case is available to use to 11000 m depth. This consists of LEDs, a copper base plate, resistors, an underwater connector, and a 1/2" clear tube (Yoshida 2007b).

c. Stereoscopic HDTV Camera System.

Three-dimensional (3-D) television is one application for a stereoscopic camera system. 3-D television would make an effective operation environment for vehicle operators and viewers. There are lots of commercial software and hardware solutions to make and display 3-D images on a television display and a television screen. Miracube C190x produced by PAVONINE INC. for presentations aimed at small groups employs a 3-D expression method called the Parallax Barrier (Meacham, 1986.). This method doesn't need the observer to wear special glasses but only a single user can enjoy 3D vision and only from certain positions. Use of commercial projector systems for 3-D vision uses shutter glasses or polarizer glasses for users. The use of HDTV cameras for 3-D television gives the audience a more realistic experience. The *PICASSO-1* vehicle has the capability to deploy a stereoscopic HDTV camera system. The configuration of the camera system is shown in Figure 3. The major part of the system consists of two pressure-tight HDTV cameras (HDR-SR7 made by SONY) and a controller. Each aluminum pressure hull (ϕ170mm x 390 mm; 9 kilograms in air; depth rating of 4,000 meters; acrylic window) includes an HDTV camera, an interface

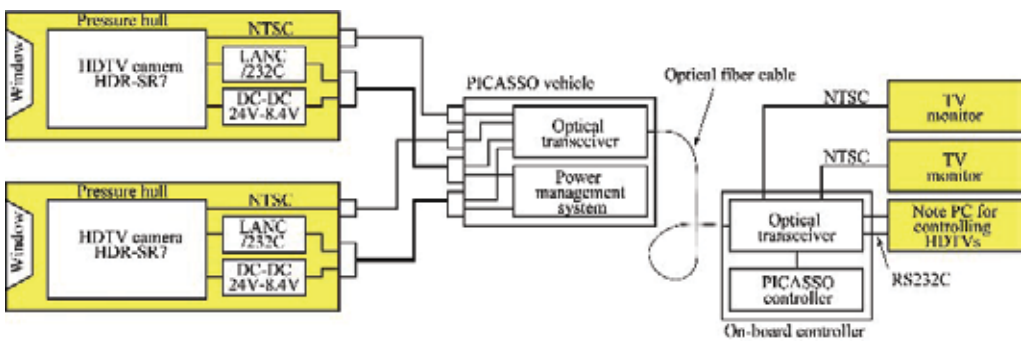


Fig. 3. System configuration of the stereoscopic high definition television camera system installed in the *PICASSO-1* system.

Fig. 4. *PICASSO-1* equipped with the stereoscopic HDTV camera system. Two LED light arrays were additionally made for this system and installed on either side.

adaptor, and a DC-DC converter. HDTV images (MPEG4 AVC/H.264) are locally recorded on the internal 60GB hard disk of the HDR-SR7. Figure 4 shows a snap shot of the *PICASSO-1* vehicle equipped with this stereoscopic HDTV camera system.

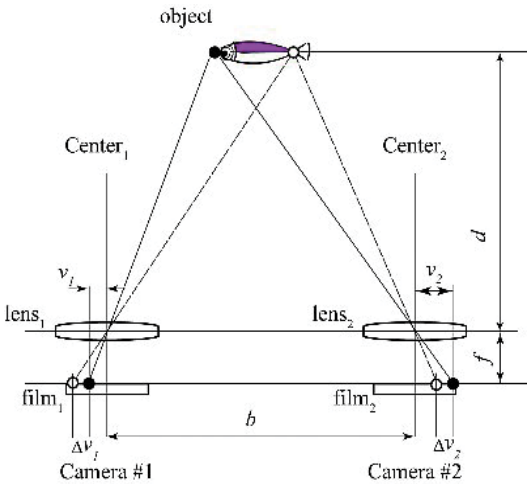


Fig. 5. Camera placement and coordinate system for stereovision.

The other application for the stereoscopic camera system is as an object scale estimation system. By using HDTV cameras for scale estimation, the resolution of the system become threefold compared with a conventional NTSC-based camera system. For measuring the distance to an object and estimating its size using stereovision, triangulation is generally used. In this method a disparity map is prepared. The disparity map is a depth map where the depth information is derived from offset images of the same scene. Figure 5 shows the coordinate system of the camera system for calculation. The disparity (d) between the left and right image points is defined as the difference between v_2 and v_1. The depth; D is calculated from equation 1,

$$D = \frac{bf}{d} \tag{1}$$

Where b, f, and d denote base offset, focal length of camera (distance between lens and film), and disparity, respectively. Object size; S is roughly estimated from equation 2,

$$s = \frac{b}{2d}(\Delta v_1 + \Delta v_2) \tag{2}$$

In this equation, Δv_1 and Δv_2 are the image size on each film. To measure disparity in the camera system, we compute a given pixel location in either the right or left image coordinate frame with a stereo matching technique. Zitnick and Kanade (Zitnick & Kanade, 1999) have developed a better stereo algorithm. For calculation in real time using high definition images, a very high performance computer would be needed, so this calculation will be done after a dive has finished.

d. Inertial Navigation System (INS)

An INS is one of the most important devices for an AUV because an AUV must obtain an accurate position and information on any attitude changes itself. IXSEA's Phins, which is an INS based on a fiber optic gyroscope having a pure inertial position accuracy of 0.6 NM/hour, is widely used with a Doppler velocity log (DVL) in AUVs. A sufficient level of position accuracy is achieved by the aid of an external sensor, a ground referenced DVL. Larsen reported (Larsen 2002) that the Doppler-inertia based dead-reckoning navigation system, MARPOS, has a proven accuracy of 0.1 per cent of the distance traveled for straight-line trajectories. If an AUV equipped with an INS/DVL hybrid system cruises at a high altitude from a seafloor, a DVL cannot measure its velocity. This leads to increase of positioning error. To reduce this error an AUV usually requires an acoustic navigation system and operators set acoustic transponders in underwater positions before deployment of the AUV. In the case of longer range AUV operations, the time period of AUV navigation using pure inertial positioning data becomes long and this means that many transponders must be deployed – usually an untenable solution. From this point of view an INS should have the highest pure inertial position accuracy possible. Ishibashi *et. al.* have proposed a unique error reducing technique based on a ring laser gyro (Ishibashi 2008). The position error of an INS results from its drift-bias errors, the sources of which are unidentified random noises. They have proposed a method where the axial rotational motion is applied to the INS. They were able to achieve a high pure inertial position accuracy of 0.09 NM/hour by this method.

e. Ultra Short Base Line (USBL) System

Acoustic navigation systems for underwater vehicles are produced by many companies but USBL systems with full depth capability are very rare. Watanabe *et. al.* (Watanabe 2006) have developed a small USBL system for full depth use. The system consists of two major parts: a USBL transceiver installed on the station and a transponder fixed on the probe. Table 1 shows the specifications of the USBL system. The accuracy of the position is relatively low because the probe position is directly obtained using the station TV camera in their plan. In this system, the M-sequence signal is used as the modulation signal. An original processing unit has been developed using a DSP (Black Fin produced by Analog devices) and an FPGA (Cyclone produced by Altera). The system was tested in the Marianas Trench in 2008.

2.2 Communications devices and methods needed for each vehicle

Optical communication systems allow operators access to high speed data delivery and allows real-time control of a vehicle. The systems are widely used for communications

Items	Specifications
Beam width	120 deg
Accuracy	<5% within 200 m range
Range	2,000 m
Depth rating	11,000 m
Frequency	20 kHz
Modulation	BPSK
Data	M-sequence signal
Sensors	Sound velocity meter
Transducer	4 array
TX sound pressure	180 dB re uPa at 1m
RX sensitivity	-210 dB re 1V/uPa at 1m

Table 1. Specifications of the USBL

solutions with ROVs, UROVs and HROVs. In recent years, data traffic on networks has drastically increased with the evolution of broadband networks. In order to meet the demand, developers are trying to develop a 40 Gbps optical communication system using a dense wavelength division multiplexing technique for land and submergible cable applications.

For wireless remote control and status monitoring of AUVs, an acoustic communication system or an acoustic modem is used. This is also effective for monitoring an UROV or an HROV. For close-range communication, electromagnetic communication would be useful because radio communication performance would be less affected by multi-pass interference. Optical communication systems having a capacity of 622 Mbps and 2.488 Gbps are generally used for underwater vehicles. Prizm Advanced Communication Electronics Inc. provides a communication board with an HD-SDI interface. Canare in Japan manufactures fiber-optic products including an 8-channel coarse wavelength division multiplexing HD-SDI transceiver module. Neither of these manufacturers produces an all-in-one optical transceiver, which would consist of video interfaces, serial data interfaces, and parallel interfaces on one printed circuit board. Yoshida *et. al.* (Yoshida 2007b) have developed two types optical communication boards: one is an optical-electrical communication system for the *ABISMO* system and the other is a high speed device for an UROV vehicle, with the prototype being installed in the *PICASSO* system.

a. An Optical-electrical Communication System

The *ABISMO* system consists of a launcher and a vehicle. The support ship and launcher are mutually connected by optical fiber cable for data transmission. The launcher and the vehicle are mutually connected by a metallic cable. Three-point-communication (the ship – the launcher – the vehicle) is therefore needed in the *ABSIMO* system. The block diagram of the optical communication system model, JT3 for the ship-launcher communication and the radio frequency digital communication device, JT3-RC for the station-probe communication, are depicted in Figure 6. Its optical communication bit rate is the same as the SONET (STM-4) standard but the protocol is an original one. Every input signal is sampled, time shared, Manchester encoded, and then transmitted at a bit rate of 622 Mbps. The JT3-RC is a full duplex transceiver with 8 RS-232C channels. In the JT3-RC circuit board, its synchronization is achieved by a sequential synchronization using Manchester encoding with a 16 bit preamble. The time-division multiplex data rate is 12.96 Mbps. Maximum transmission range is designed to be 200 meters by using 2.5-2 V standard coaxial cable. A pre-emphasis

circuit reduces deformation of the transmission wave caused by loss through the cable. This system was practically tested in the Marianas Trench in June 2008 at a depth of 10300 m.

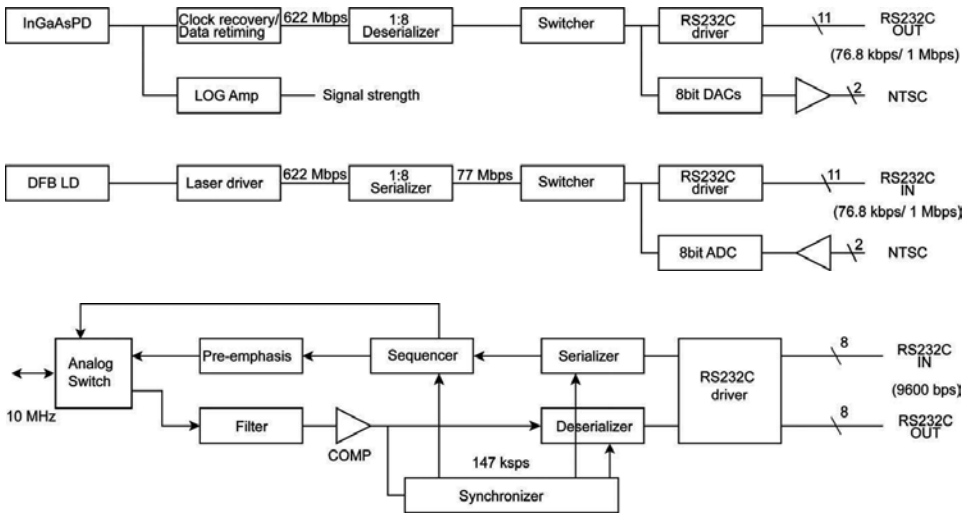


Fig. 6. The block diagram of the optical communication part of the JT3 (upper) and the blockdiagram of the JT3-RC. The synchronizer in JT3-RC regenerates the sampling clock.

b. A Low Cost 2.5 Gbps Optical Communication System with HD-SDI Interface

The system consists of a pair of transceiver units for the vehicle and the ship side. The transceiver unit consists of two printed circuit boards: a protocol converter board and a power supply board (each board size is 120 x 80 mm). Major devices for the converter are a 2488 Mbps optical transceiver module produced by Sumitomo Electric Industries, Ltd. and a TLK3101 transceiver chip by Texas Instruments Incorporated which is composed of 2.5 Gbps to 3.125 Gbps Serializer / Deserializer. The transceiver has the interfaces: one HD-SDI data interface for an HDTV camera, three NTSC interfaces, four RS-232C interfaces, two RS-485 interfaces, and 8-channel parallel I/O interfaces.

c. Acoustic Modem Using Time-Reversal Waves in Shallow Water

An advanced acoustic communication method utilizing time-reversal waves has been developed (Kuperman 1998, Shimura 2004). In most acoustic communications the ship-vehicle configuration is vertical because there are many multi-path signals in the horizontal configuration. It would be better to use a time-reversal technique for communication under multi-path fading in the shallow water zone. Shimura did a simulation for communication between a ship and a vehicle in the shallow water zone using high frequencies (Shimura 2006). He reported that the method of time-reversal process with an adaptive filter provides good communication results. When the vehicle, however; moves, the advantage of the method is depressed. We will try to modify the method and choose the best parameters, aiming at better ship-vehicle communication up to 500 m in distance.

d. Communication by Electromagnetic Field.

In seawater the attenuation coefficient, α in the HF band and below is obtained by equation 3 which is derived from Maxwell equations.

$$\alpha = 8.686 \times \sqrt{\pi \mu_0 \sigma_0} \sqrt{f} \ (\mathrm{dB/m}), \tag{3}$$

where μ_0 is the permeability, σ_0 is the conductivity of the seawater, and f is frequency in Hertz. Substitution of $\mu_0 = 4\pi \times 10^{-7}$ and $\sigma = 4$ S/m into equation 3, one obtains,

$$\alpha = 3.45 \times 10^{-2} \sqrt{f} \ (\mathrm{dB/m}). \tag{4}$$

The equation means that an RF wave in seawater is rapidly damped, for example 128 dB/m at 10 MHz. A number of tries at RF communication in seawater have been made. Siegel attempted propagation measurements in seawater at 100 kHz and 14 MHz (Siegel & King 1973) by preparing a special underwater antenna. They concluded that the experimental data are in good agreement with theoretically obtained data from asymptotic formulas. A new approach to electromagnetic wave propagation through seawater has been proposed (Al-Shamma'a 2004). In their theory, there are conduction currents in the near field and displacement currents in the far field. This causes rapid signal attenuation in the vicinity of the antenna but in the far field the attenuation is comparable with the dielectric loss. JAMSTEC has also carried out propagation measurements in seawater from a quay. The propagation characteristics in the ELF roughly agreed with the theoretical characteristics. The curve according to the HF measurement data as shown in figure 7 is similar to the one that Al-Shamma'a obtained. This means that someone should make a careful investigation at HF.

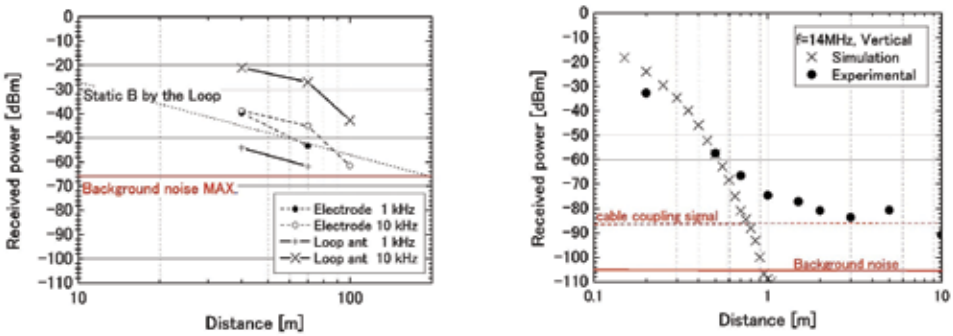


Fig. 7. Propagation characteristics of electromagnetic waves in seawater in the ELF band (left) and the HF band (right).

JAMSTEC has been developing a new communication tool that uses electromagnetic waves. This method is used for mutual communication between vehicles at up to 50 m distance. A prototype transmitter, a receiver, and antennas were made. An NTSC camera for underwater use was connected to the transmitter. The transmitter encodes and modulates the image data and then supplies power of 17 Watts to a multi-turn coil antenna. A high sensitivity search coil antenna receives the modulated data. The receiver demodulates, decodes, and outputs the image in QVGA format. In the tank test, QVGA images were transmitted to the receiver set 30 m away from the transmitter.

e. Satellite Communication system

Most satellite communications from the ocean use an earth orbiter satellite, for example Argos satellites and Iridium satellites, rather than a geostationary satellite because the latter needs a large sized antenna such as a parabolic antenna. However, a geostationary satellite can provide full real-time communication and a large coverage area. The Eighth

Engineering Test Satellite, ETS-VIII has as its main purpose, dealing with the increasing demand for digital communications, such as mobile phones and other mobile devices. The satellite (weight of 3 tons and diameter of 40 m) has two Large Deployable Antenna Reflectors (LDARs). Table 2 lists specifications of the ETS-VIII.

Fig. 8. Picture of ELF wave transceiver tested in tap water and a received image.

Figure 9 shows the concept of this project. Devices or facilities on the ocean are able to communicate with land stations via a satellite with a large capacity wireless network. This enables us to remotely control these devices or facilities, resulting in effective research in marine-earth science. By using the ETS-VIII, remote control of an underwater vehicle from a land station at JAMSTEC as shown in Figure 9 will be possible. For this purpose development of a satellite communication system with help from the Japan Aerospace Exploration Agency and National Institute of Information and Communications Technology was started since 2003.

Items	Spec.	Unit	Remarks
Downlink freq	2500.5-2503.0	MHz	
Up link freq	2655.5-2658.0	MHz	
Satellite EIRP	61.8-63.8	dBW	
Satellite G/T	12-14	dB/K	
Satellite Antenna Gain	41	dBi	
Communication rate	64 - 384	Kbps	internal ant

Table 2. The specifications of ETS-8.

A custom antenna is needed for underwater vehicles because there is no commercial pressure-tight or water-resistant small antenna. A left-handed circularly polarized; double resonance antenna is matched to the ETS-VIII. Antenna minimum gain is 6.3 dBi. A four-element patch antenna, with a gain of about 14 dBi, and a single element patch antenna with phase difference feeding lines were made. The gain of the single element planar antenna is only 7 dBi. By decreasing the communication rate by 64 kbps, which gains 7 dB on power per bit compared with 384 kbps, antenna margin is kept.

The ocean-based system should be equipped with a satellite tracking system to lock on to the satellite, because of the vehicle oscillation. The tracker must also be water-resistant. For these reasons, a tracking system has developed using data on network design and the oscillating characteristics of the *Urashima* vehicle. Oscillation of the vehicle was measured with the high accuracy inertial navigation system installed in *Urashima* in a sea trial. An oscillation angle of 7 degrees and period of 0.15 Hz were estimated. We set the design target to pitching and rolling angles of less than 20 degrees and maximum frequency of 0.5 Hz, The tracking system consists of an attitude sensor and an attitude controller. To obtain the direction of the satellite, an inertial navigation system and a GPS are used as attitude sensors. The attitude controller has a three axis stepping motor driver. The system is currently undergoing tests with the first sea trial in November 2008.

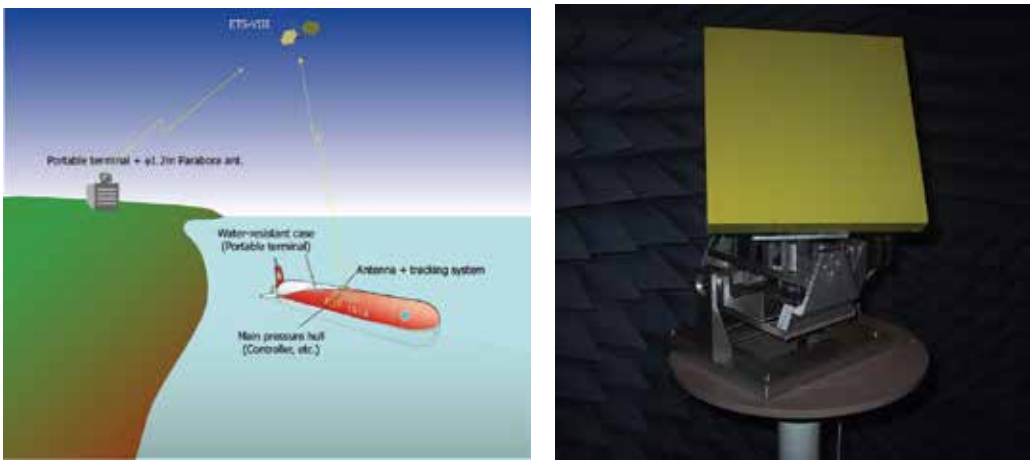


Fig. 9. (left) An image of satellite communications with an underwater vehicle. (right) Photograph of the attitude controller with the 4-element planar antenna.

2.3 Modern power sources

Power sources are extremely important in underwater vehicle development, in particular for AUVs, UROVs and HROVs. Power source capacity limits the cruising range and mission style of underwater vehicles. Two main evaluation factors for power sources are the specific energy, energy per unit mass: Wh/kg, and the energy density, energy per unit volume: Wh/L. In vehicle design, not only the energy of the power source is considered, but also the maximum output power. In this section, modern power sources for intelligent underwater vehicles, secondary batteries and fuel cells in particular are described because these are rechargeable and are able to be run in a closed system in the underwater environment.

a. Batteries

A number of kinds of secondary battery, lead-acid, silver-zinc, nickel-MH, lithium-ion and lithium-polymer batteries are utilized in AUV design. Lithium-ion and lithium–polymer batteries have the advantages of easy handling, higher energy density and longer cycle life as shown in Table 1. Davies and Moore (Davies 2007) have proposed the ratio of specific energy and energy density, D as an index for helping power source design.

$$D \text{ (kg/L)} = \text{energy density} / \text{specific energy} \tag{5}$$

If D is smaller than the density of seawater, approximately 1.03 kg/L, a battery system has positive buoyancy. Calculating the D of various batteries from table 1, it can be seen that lithium type batteries have a smaller density than other types. We thus focus on lithium-ion batteries and lithium-polymer batteries because they have good prospects for future use and development (Armand 2008). Both batteries use lithium metallic oxide in the cathode and carbon material in the anode. Lithium-ion batteries use lithium ions in an electrolyte inside the battery and these transfer between the cathode and the anode during charge or discharge. In contrast lithium–polymer batteries use a solid polymer composite. The advantages of Li-polymer over the lithium-ion design include lower cost of manufacturing and being more robust to physical damage.

Battery Type	Specific energy, Wh/kg	Energy density, Wh/L	Cycle life
Lead-acid	20-30	60-80	700
Silver-zinc	100-120	180-200	100
Nickel-MH	50-70	100-150	1500
Lithium-ion	90-150	150-200	600-1000
Lithium-polymer	130-190	170-240	300-3000

Table 3. Performance of batteries (ref. Abu Sharkh 2003)

Lithium type secondary batteries have been used in a number of AUVs including Autosub6000 (McPhail 2007), ABE (Bradley 2000), Nereus (Bowen 2004), MMT3000 (Gornak 2006), and *Urashima* (Aoki 2001). REMUS, which is a mass-produced compact AUV developed at WHOI and is now commercially available only through Hydroid, Inc., also uses lithium ion batteries. Autosub6000, 5.5 m long and 2000 kg in weight, runs over 1000 km at 1 m/s and is powered by 12 pressure balanced Lithium-polymer battery packs including Kokam cells. Each pack stores energy of 18 MJ within 405 battery cells. The Urashima vehicle, designed by JAMSTEC, is powered by two pressure balanced lithium-ion battery packs: a main battery pack of 15.6kWh of energy and a 3.6 kW sub battery pack. It has travelled over 120 km and the energy density of its batteries is about 180 Wh/L. JAMSTEC has investigated lithium-ion battery performance by changing the cathode material and battery shape. They have now obtained a sheet type lithium-ion battery with an energy density of over 210 Wh/L. A pressure resistance test using an oil-filled pressure balance case was done up to 11000 m in depth.

Utilization of lithium-based batteries will continue for a considerable period of time in the future because most small AUV designers will choose higher energy density batteries and a vehicle mounting a generator requires a battery for start-up. The nanotechnology revolution will help increase the performance of lithium ion batteries in terms of capacity, power, cost, and materials sustainability (Armand 2008) in the near future. Lithium-oxygen batteries, which can have a capacity of 1200 mAh/g according to the reaction 2Li + O2 -> Li2O2, have greater potential compared with lithium-ion batteries of about 150 mAh/g, theoretically. There are now prominent failures in this type of battery but Armand expects that much more work may break through the issues after 2050. If this battery becomes of practical use, the cruising range of every AUV will see an eightfold increase from that of present AUVs.

b. Fuel Cells and Semi-Fuel Cells

A semi fuel cell is a generator but its usability is rather like a battery because it requires a reactant, hydrogen peroxide, besides an exchange of the anode, due to corrosion of the aluminium cathode in the electrical generation process, and the electrolyte. Aluminium/hydrogen peroxide energy semi-fuel cells can theoretically generate an energy density of 3418Wh/kg and practically one of about 400 Wh/kg. This corresponds to 3 times that of a lithium-ion system. This type of semi-fuel cell contains only liquid and solid materials, independently running under the ambient pressure. A vehicle designer is thus able to design a pressure balanced battery system with a semi-fuel cell. A semi-fuel cell would be suitable for mid size underwater vehicles because the size of a pressure-hull-less semi- fuel cell is not so large (Adams 2002). The Hugin 3000 autonomous underwater vehicle, 5 m long and 1400 kg in weight, uses a semi fuel cell as the main power source (Hasvold 2002). This semi-fuel cell generates energy of 45 kWh for 50 hours.

Many types of fuel cell system have been developed around the world. Proton exchange membrane fuel cells (PEMFC) are the most suitable for underwater applications such as for autonomous underwater vehicles. Its operation temperature are around 70 degrees Celsius and its reactive product is only pure water. Underwater, a typical PEMFC system for land applications, such as found in automobiles, cannot be used because intake air does not exist underwater and the water reaction product is not easily drained into the high pressure external environment. The underwater vehicle *Urashima* is equipped with a closed-cycle PEFC system that consists of a fuel cell generator, high pressure oxygen tank, and a metal hydride tank. Its generating system must be perfectly closed so that there is no emissions underwater. The energy density of the fuel cell generator itself is high, although for the whole fuel cell system has a lower value due to weight gain from hydrogen, oxygen, and reactant water tanks, auxiliary components, and control electronics. Decreasing the size and weight of these devices is needed for underwater applications of fuel cell technology.

JAMSTEC has developed underwater vehicles for surveys in the vast underwater environment. The vehicles are utilized for sea floor observations, ocean environmental research, energy source exploration, and research on marine organisms and micro-organisms. One of the important underwater vehicles is an AUV with a large capacity energy source, a highly accurate positioning system, and a smart control system for autonomous cruising. In 2005, JAMSTEC made a world record of cruising distance of 317 km by the autonomous underwater vehicle *Urashima*, powered by a closed-cycle PEFC system. They aim to develop an underwater platform that can survey across entire oceans for scientific research into global climate change, ocean-trench earthquakes, marine microorganisms and multicellular organisms. In 2007, they started research and development on a second generation long-range cruising AUV (LCAUV) to cruise over 3000 km. The development of an improved power source for the vehicle is important to realize this goal. A fuel cell system has to be the best choice for a power source aimed at long-range cruising with a limited payload.

The PEFC system for the LCAUV must satisfy the following requirements; 1) high efficiency (over 60 %), 2) fuel of pure hydrogen and oxygen and downsizing of the storage system, 3) leakless stacks, 4) perfectly closed system, 5) over 600 hours continuous running time (need high reliability and durability), and 6) small system. Table 1 shows the power system specifications required of successive LCAUVs.

Term	Platform	Endurance	Range	Power	capacity
1998 - present	Urashima	60h	300 km	4 kW (Max)	180 kWh
2007 - 2015	2nd LCAUV	600h	3,000 km	10 kW (Max)	5000 kWh
2016 or later	3rd LCAUV	?	10,000 km	?	?

Table 4. Power system specifications

Urashima, the first prototype of a fuel cell-driven underwater vehicle built by Mitsubishi Heavy Industry Ltd., has the following specifications: length; 10 m, weight; 10 tons, maximum depth rating; 3500 m, maximum cruising speed; 3.2 knots, and endurance; 60 hours. Fuel cells for underwater vehicles should run on pure-hydrogen and pure-oxygen since no air exists underwater. The water byproduct produced in the fuel cell should be stored in the vehicle body to keep its buoyancy constant. If the reactant water is pumped into the external environment, the vehicle consumes much more energy and the vehicle loses weight and will start to float. We have thus developed a completely closed fuel cell

system, which confines energy resources and reactant water to the system, namely the closed-cycle PEFC system as shown in Figure 10. This FC system consists of two stacks, recirculation blowers, humidifiers, a heat exchanger and a reactant water storage tank, generating power of 4 kW. All devices are installed into a titanium pressure vessel. The coolant water from the FC stack is reused to humidify the hydrogen gas. A metal hydride (MH) vessel and a high-pressure oxygen tank are included. The heat generated in the FC stack is applied to heating the MH to extract hydrogen from the MH and the excess heat is radiated into seawater. Figure 11 shows a typical I-V plot of the PEFC system obtained in the 317 km sea trial. The maximum FC system efficiency was about 54 % at typical cruising speed.

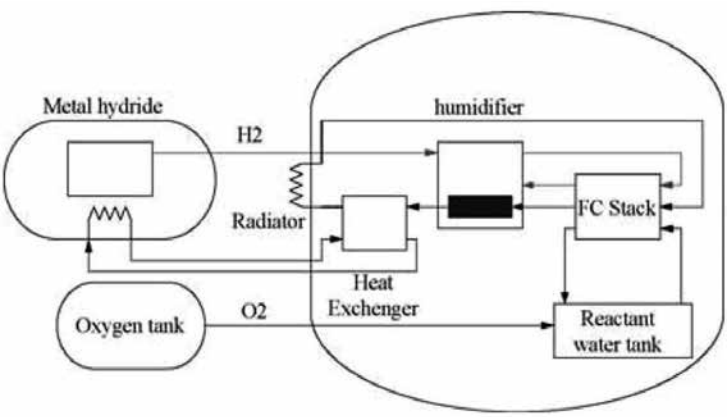


Fig. 10. System configuration of the closed-cycle PEFC for *Urashima*.

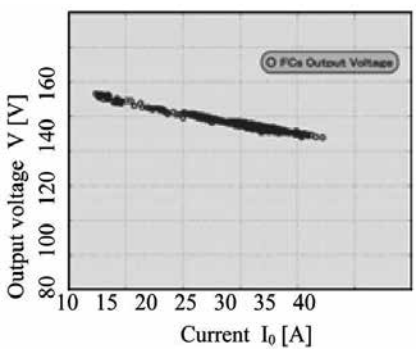


Fig. 11. I-V plotting of the closed-cycle PEFC during a 317 km.

The target cruising range of the 2nd LCAUV has increased tenfold from that of the *Urashima*. JAMSTEC has set target specifications for the fuel cell system, with system efficiency of over 60 % and downsizing of the pressure vessel and tanks as shown in Table 2. They made and tested a single cell which consists of a solid polymer electrolyte membrane, carbon black, platinum-alloy, carbon paper, and metallic separators. The test was done under the following conditions: cell temperature of 60 degrees Celsius, process pressure of 300 kPaA, and gas utility factor of 50 %. Figure 12 shows an I-V plot of the new cell compared with the *Urashima* system. In the figure the circle and the square show that of *Urashima* and that of the new cell, respectively. They have a single cell efficiency of 60 % at a higher heating value

in the target current point. Now JAMSTEC has designed a blower- and humidifier-less system using the new cells.

Platform	Efficiency	Volume ratio
Urashima	54 %	1
2nd LCAUV	60 %	1/2 or less

Table 5. Required PEFC specifications for underwater vehicle

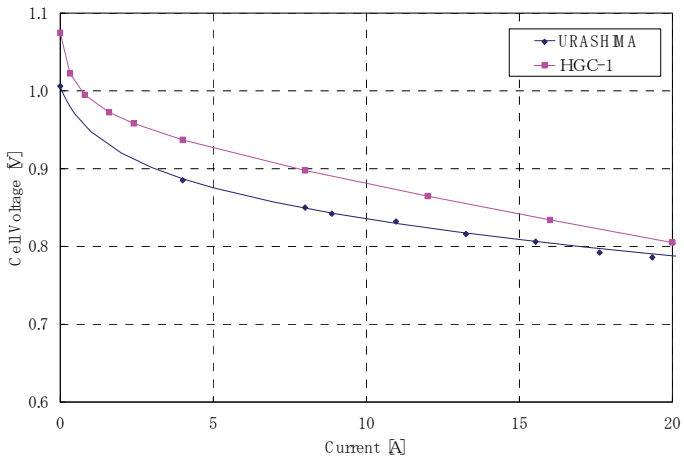


Fig. 12. I-V curve of new cell

3. Data processing

3.1 Control hardware: internal communication bus and distributed CPU system

In control hardware robustness, reliability, high speed, synchronization are important. A distributed CPU system reduces concentration of processing load on a single CPU and allows system redundancy. A distributed CPU system can be composed of printed circuit boards with an embedded CPU chip and an internal bus. In the distributed real-time system, it is important which internal bus is the best for the system considered. Some researchers (Weidong 2006, Blandin 1998, Yoshida 2004) proposed the the Controller Area Network (CAN) bus which was originally developed in the 1980's by R.BOSCH GmbH as an internal bus for AUVs. The CAN bus is based on the broadcast communication mechanism. Every message has a message identifier, which is unique within the whole network since it defines content and the priority of the message. The CAN bus also has the mechanism of bit and frame synchronization. The maximum data transmission rate of CAN is 10 MHz.

3.2 Image sensing and recognition

a. Midwater Organism Tracker

PICASSO will semi-automatically track animals in midwater. In order to detect and track an animal the vehicle has to incorporate animal image recognition and then automatically move so as not to lose the animal that has been recognized. JAMSTEC has developed a prototype system for an animal tracker using the *MROV* vehicle. To simplify the prototype system, only the pan-tilt system of the camera rather than the entire vehicle itself was

controlled by the tracking program. Color deference in HLS (Hue, Saturation, Luminance) color space was basically used for detection. Identification of a target is initially done by clicking on the target on the display. RGB values in the 9 x 9 pixels around the pixel clicked are converted to the HLS color space. A center of gravity for pixels with the near-HLS value obtained is then calculated. When the distance between the center of gravity and the center of the image obtained by the camera exceeds a preset limit of the pan-tilt, the program controls the camera to center the animal in the middle of the observation space. The program also has a displacement prediction function to predict movements of animals.
A detection and tracking test was carried out in the large fish tank (6.5 m in depth, 144 m^2 area of base) at the Enoshima Aquarium. A scene taken during the test is shown in Figure 13. The prototype system was able to detect and track a small fish (red circle in the figure) for 30 seconds in this test. However, in most cases the duration of capturing the target was only a few seconds because there were many fish in the tank and the background-target contrast was low compared to in the midwater zone of the ocean. For more accurate detection, they will collaborate on an image recognition method with MBARI (Walther 2004). This method simulates human vision functions and has a high target recognition probability. JAMSTEC will also investigate a program to track animals by linking this output with thruster control.

Fig. 13. A tracking test using the *MROV* in the Enoshima Aquarium. The cross shows the tracking point.

4. Present intelligent underwater vehicles

In this section, vehicles, equipped with state-of-the-art devices, that were developed at the institute for which the author works are the mainfocus.

4.1 Plankton survey vehicles

Research on planktonic organisms is important because they are the link between greenhouse gases being absorbed by the ocean and the final burial of these gases as solid organic carbon in deep sea sediments. Planktonic organisms also occur at very high point biodiversities and insights into how so many species can co-exist in a seemingly homogeneous environment should help shed light on aspects of biodiversity that need to be grasped for protection of biodiversity hotspots and to understand evolution.

Several trials with ROVs and manned submersibles (Wiebe & Benfield, 2003) have been carried out to investigate the distributions of macro- and micro-plankton versus environmental parameters. In this way, one is only able to gain information of a point nature and it is not possible to determine large-scale distributional patterns with limited ship-time. Both winch-controlled towed systems (MOCNESS net, BIONESS net, BIOMAPER-II system) have been equipped with a combination of imaging, acoustic and environmental parameter sensors. However, the maximum operation depth for the BIOMAPER-II and SeaSoar were only 300 m and none of these systems had imaging systems of high enough resolution to identify and quantify plankton at the species level (Wiebe & Benfield, 2003).

Since 2005, JAMSTEC has been developing a multiple-platform autonomous survey system able to quantitatively characterize the midwater environment, including fragile components such as large particulates and gelatinous plankton. This system could be deployable from small to medium sized boats and ships.

Since 2006, we have developed the first small vehicle named *PICASSO-1* Plankton Investigatory Collaborating Autonomous Survey System Operon-1. Fig. 14 shows a snap shot of *PICASSO-1* during a sea trial. *PICASSO-1* is small and light (2.4 m long, 200 kg in weight) and the color of the hull is mostly red because deep sea organisms cannot see light or reflections in the red spectrum as a rule. The vehicle system consists of an on-board topside module and a vehicle, and these are connected via a thin optical fiber cable. One remotely controls the vehicle from the topside module. *PICASSO-1* is composed of the following major parts: an FRP fairing cover, a body frame, buoyancy materials, controllers, communication systems, three 100 W thrusters, one tilt actuator, lights, devices for navigation and observation, oil-filled lithium ion battery, and an optical fiber spooler. The vehicle has one vertical tail fin and two fins for stability. Table 6 shows the PICASSO-1 specifications.

Item	Specification	Remarks
Dimension	2.1 m x 0.8 m x 0.8 m	without VPR
Weight	200 kg	in air
Depth rating	1,000 m	
Cruising speed	2 kt	
Endurance	6 hours	
Operation mode	UROV	
Propulsion	2 horizontal 100 Watt thrusters with tilt system, 1 vertical 100 Watt thruster	
Communications instrumentation	2 G bps optical communication device , Radio LAN, ARGOS transmitter, acoustic and magnetic transceiver*.	
Navigation instrumentation	MEMS gyro, Doppler velocity log, depth meter, SSBL, compass	
Experiment payload	CTD, TDO, Fluorometer-Turbidity sensor, 4 x NTSC cameras, 3 x 35 Watt HID lamps, Visual Plankton Recorder*, High definition TV camera*, Digital still camera*, 400 watt HID lamp*.	* pick only one device

Table 6. Specifications of *PICASSO-1*

One of the purposes for development of the system is to track plankton. A typical swimming velocity for plankton is slow, a few hundred meters per hour (Matsumoto 1991), and its direction of swimming is random. *PICASSO-1*, therefore; has two lateral thrusters with a 180 degree tilter and a vertical thruster for highly maneuverable cruising and its maximum cruising speed is set to 2 knots. For smaller to larger animal observations, *PICASSO-1* is able to select as its main imaging tool from among three choices: an HDTV camera, a 12 bit high resolution camera, and an underwater "microscope" called the color Visual Plankton Recorder (VPR).

b. The sea trial of *PICASSO-1* in February-March 2007

A sea trial was carried out in 24 February – 4 March in Sagami Bay and Suruga Bay using the support vessel, *Natsushima*. Seven dives were made. In two dives, the VPR was installed on *PICASSO-1* (Figure 14). The vehicle dived to a maximum depth of 601 meters. All function tests were performed well. The images obtained by the HDTV camera were of extremely high quality and several animals were observed. Meso zoo plankton were observed by the VPR (Figure 15). This first sea trial of *PICASSO-1* was successfully finished. Up to the present date *PICASSO-1* has made 25 dives, including dives using a small support ship (Yoshida 2008).

Fig. 14. *PICASSO-1* with the Visual Plankton Recorder being recovered.

Fig. 15. An undescribed species of Bathyctenid comb jelly and a Gnathophausid mysid shrimp obtained by the Visual Plankton Recorder installed in *PICASSO-1*.

4.2 Automatic sediment sampling ROV at full ocean depth

The *ABISMO* system and its 11,000m-cable store winch are mounted on the dedicated ship *Kairei*. The *ABISMO* system consists of an on-board control module which is installed in the ship, a sampling station, a sediment probe, and two samplers. Fig. 16 shows a recovery scene of the station housing the probe. The on-board control module is connected with the station via the primary cable. The secondary cable, which has been newly developed, connects the station and the probe. The sampling station houses the probe and one of the samplers in the bottom cage. The station is mounted with a docking-undocking system and a secondary cable drum for the probe, sampler release gear and a rope-hoisting winch for the sampler. The station furthermore serves as a repeater between the on-board equipment and the probe. The probe cruises below the station freely within the reach of the 160 m cable to survey the sea-bottom surface with a TV camera. The probe is able to take a small sediment sample with a mini manipulator. Two types of sediment samplers - a gravity core sampler and a grab bottom sampler have been prepared. Scientists can choose either sampler in accordance with the intended use. The system specifications are shown in Table 7.

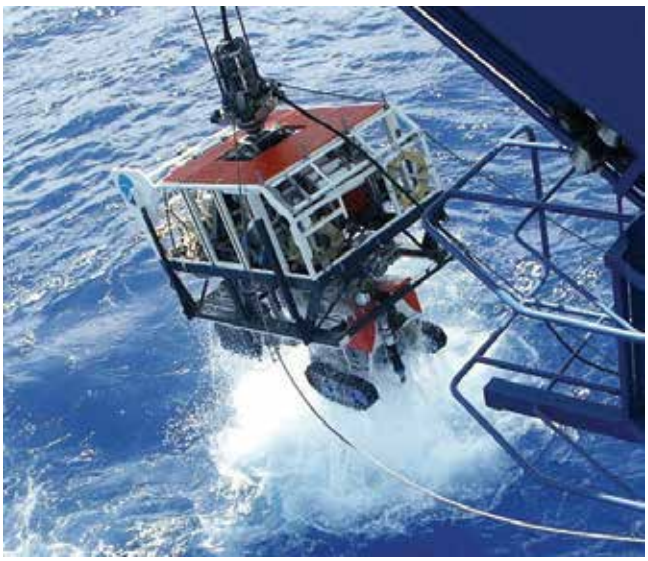


Fig. 16. The full depth ROV, *ABISMO*

Item	Launcher	Vehicle
Rated Depth	11,000 m	11,000 m
Dimensions	3.28 x 2.09 x 1.76(2.78*) m	1.22 x 1.30x 1.215m
Weight in air	3,070 kg	327 kg
Weight in water	2,300 kg	97 kg
Depth rating	11000 m	11000 m

* overall height including the vehicle

Table 7. Specifications of *ABISMO*

a. The first sea trial of the *ABISMO* system in January 2007

The first dive was made at Yokosuka 4th district in Tokyo bay to check system function on January 5. The test results were good and the ship then headed to Sagami Bay. During the second dive an assessment was made as to whether the station thruster could constrain its self rotational motion caused by the primary cable twisting or not. The thrusters behaved well during the 200 meter dive. On January 8 we tried sampling with the gravity core sampler at a location where the bottom sediment was softish, at a depth of 480 m. The station was controlled keeping its heading, coming as close as about 80 meters off the bottom. Keeping its altitude, the sampler was dropped. The sampler was recovered after about ten minutes. A sediment sample about 200 mm long then was obtained.

b. The successful sea trial in the Mariana Trench in June 2008

The fourth sea trial including dives in the Mariana trench had been made from 26th May to 8th June 2008. Before going to the Mariana trench, one dive at Sagami-Bay and also one dive in the Izu-Ogasawara trench were conducted to confirm the additional functions and also the functions that had not been tested in the previous sea trials. In the Mariana Trench *ABISMO* made three dives within three days to depths below 10,000m and marking a 10,257m dive. This depth was limited by the length of the primary cable. The 2-m long gravity core sampler was dropped down in free fall and sediment cores of 1.6 m length were obtained (Figure 17). *ABISMO* also succeeded in obtaining 12 bottles of water samples from the Mariana trench in each dive.

Fig. 17. Mud sample obtained from the sea floor in the Mariana trench (2m-core sampler)

4.2 Cutting edge autonomous underwater vehicles

The prototype AUV *Urashima* is an LCAUV with a range over 300 km, as shown in Figure 18. Its specifications are listed in Table 1. The vehicle consists of titanium frames covered with FRP (Fiberglass Reinforced Plastics) faring covers, pressure vessels made of titanium alloy for protecting control systems and other electrical devices, and buoyancy materials used for additional buoyancy. The vehicle has a cylindrical shape for reducing hydrodynamic drag. The vehicle is equipped with six actuators: a main thruster (D.C. brush-less motor, 1.5 kW) for cruising, two vertical thrusters, one horizontal thruster at the stern, a vertical rudder, and a horizontal rudder. The vehicle also has a pitch control system like a level adjuster and buoyancy control system which consists of an oil tank contained in a pressure vessel (VBT: Variable Ballast Tank) and an oil bladder. The system is able to change buoyancy from 0kg to 60kg according to water depth. The vehicle is powered by a fuel cell-battery hybrid power system as mentioned.

Fig. 18. Deep & Long range Cruising AUV, *Urashima.*

Item	Specification
Dimension	10.6 m x 2.55 m x 2.4 m
Weight in air	10,000 kg / 75,000 kg without the FC system
Depth rating	3,500 m
Cruising speed	3 kt
Endurance	60 hours
Navigation instrumentation	Inertial Navigation System, Doppler Velocity Log, Acoustic Homing Sonar, Obstacle Avoidance Sonar
Experiment payload	Side Scan Sonar, Multi Beam Echo Sounder, Sub Bottom Profiler, TV camera, CTDO, Digital Camera

Table 8. Specifications of the *Urashima* vehicle

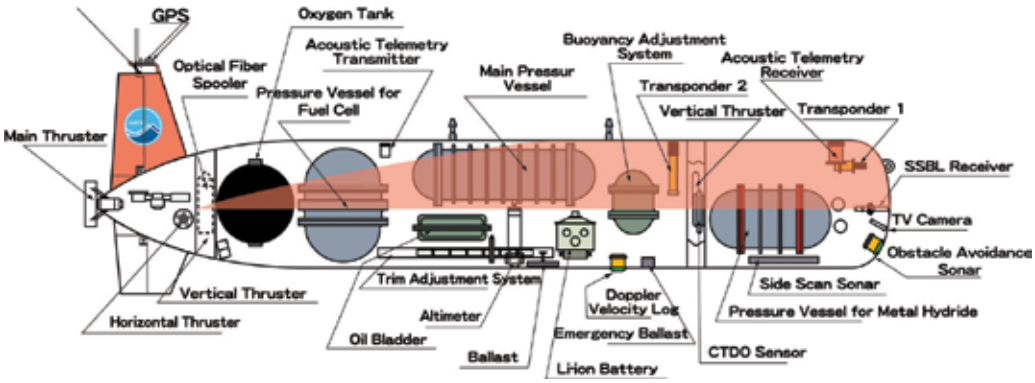


Fig. 19. General arrangement of *Urashima* with the Fuel Cell

The *Urashima,* the biggest AUV in the world, has a large payload capacityof about 500kg. Regular equipment for observations are a TV camera, digital still camera, a CTDO, a side scan sonar, a multi-beam echo sounder and a sub-bottom profiler. The side scan sonar, which has maximum range of 500 m, is a dual frequency type that projects acoustic waves at 120 and 410 kHz at central frequency. The sub-bottom profiler is a chirp sonar that uses frequencies from 1 to 6 kHz, having maximum range of about 50 m. A receiver in an array with 6 channels can record phase data. The multi-beam echo sounder observes bathymetry. The sounder uses 400 kHz burst acoustic waves,ranging about 300 m. The sounder works in a stand-alone mode. The control parameters of the sounder is set before a deployment.
The development project for the *Urashima* is started in 1998. The vehicle powered by a lithium-ion battery system has had 10 sea trials and 40 dives by 2002. In these sea trials, the vehicle both achieved a dive to 3,518m in depth and cruised a distance of 132.5 km in autonomous navigation mode over 29 hours.
In 2003, the lithium-ion battery system was replaced with the fuel cell system and then tests of the fuel cell vehicle started. In February–March 2005, the vehicle achieved a cruising distance of 317km under autonomous navigation mode. The average cruising speed was 2.8 knots. The vehicle maintained a cruising depth of 800m. The performance of the fuel cell system was good throughout the test. Cruising time was 54 hours.
Figure 20 shows the track which measured by the INS at survey on a mud volcano obtained in July 2006 in the Kumano trough. The maximum depth of the survey area was almost 2,100m. Cruising speed was 2.5 knots. Although the vehicle sometimes had almost 30m differences from programmed altitude during rapid slope angle variations, the vehicle kept a stable altitude at approximately 80m above the seafloor while cruising in autonomous mode. The difference is 0.5m between the programmed course and the obtained course by the INS. The vehicle came to within 20m of the seafloor, and cruised at slopes up to 35 degrees. Cruising distance was 28km and cruising time was 8 hours. Figure 19 also shows an acoustic image of the mud volcano based on data obtained by the side scan sonar. The resolution is very high so that the details at the top of the mud volcano can be observed.

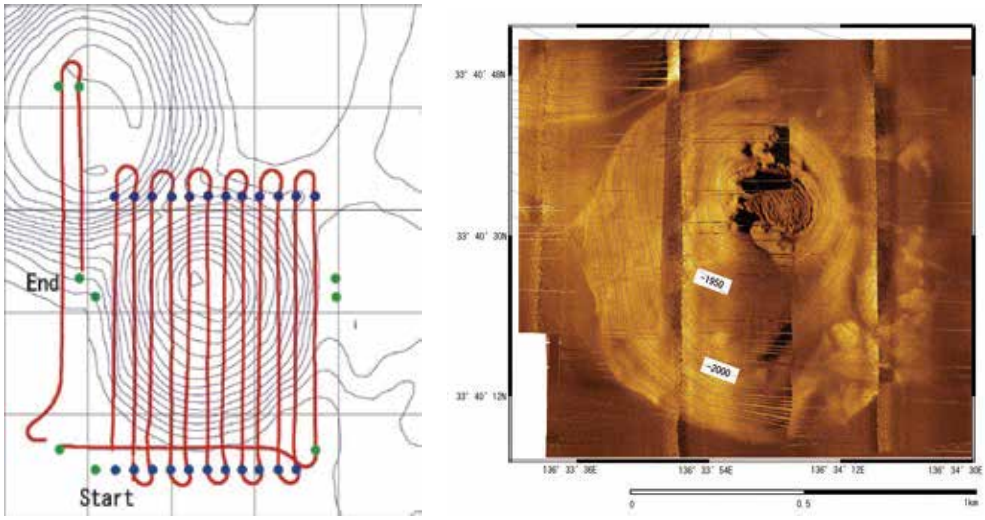


Fig. 20. Track of the vehicle during the survey of the mud volcano (left) and an acoustic image of a mud volcano obtained by the side scan sonar (right).

7. Conclusion

This chapter mainly presents information of hardware devices utilized in the development of new underwater vehicles. Basic devices including a stereoscopic HDTV camera system, communication devices and methods under development, modern power sources, and data processing methods are described. Power sources are extremely important in underwater vehicle design. The recent trends of Lithium-ion batteries, which are better for small to midsize vehicles design, and fuel cells for large vehicles are introduced. Three vehicles developed in JAMSTEC incorporated the mentioned devices and their sea trial results are shown. The development purpose of these vehicles is different but the techniques and the devices were shared in the development of each vehicle. As was mentioned in the introduction, state of the art underwater vehicles will enable a whole ocean research. The configuration of multiple deployment small AUVs and a large LCAUV may be an effective operation style in the future. The improvement of fundamental devices is essential to realize this goal. In order to improve the survey of climate change, assessment of earthquakes, and ocean resources, the accelerated development of intelligent underwater vehicles is also expected.

8. References

Abu Sharkh, S. M.; Griffiths, G. & Webb, A. D. (2003), Power Sources for Unmanned Underwater Vehicles, In: *Technology and Applications of Autonomous Underwater Vehicles,* Griffiths, G. Ed., pp. 19-35, Taylor & Francis, ISBN 0-415-30154-8, London.

Adams, M. & Halliop, W. (2002). Aluminium Energy Semi-Fuel Cell Systems For Underwater Applications: The State Of The Art And The Way Ahead, *Proceedings of OCEANS '02. MTS/IEEE*, Vol. 1, pp. 199- 202, ISBN: 0-7803-7534-3, Biloxi, Mississippi, 29-31 Oct. 2002.

Al-Shamma'a A. I.; Shaw A. & Saman, S. (2004). Propagation of Electromagnetic waves at MHz Frequencies Through Seawater, *IEEE Trans. Antennas and Propagation*, 52, 11 pp. 2843-2849, 2004.

Aoki, T.; Tsukioka, S.; Hattori, M.; Adachi, T.; Ietsugu, N.; Itoh, T.; Nakae, T. (1992), Development Of Expendable Optical Fiber Cable ROV "UROV", *OCEANS '92. 'Mastering the Oceans Through Technology'. Proceedings.* Volume 2, pp. 813 - 818, ISBN 0-7803-0838-7, Newport, Rhode Island, October 26-29,1992.

Aoki, T.; Nakajoh, H.; Nakamura, T., Ochi, H.; Tamura K.; Murashima, T.; Tsukioka, S.; Hyakudome, T.; Ida, T.; Akazawa, K. & Hashimoto, K. (2001). Deep and Long Cruising AUV Hybrid-powered with Li-ion Battery and Fuel Cell, *Underwater Intervention Conference Proceedings*, 2001.

Aoki, T.; Tsukioka, S.; Yoshida, H.; Hyakudome, T.; Ishibashi, S.; Sawa, T.; Ishikawa, A.; Tahara, J.; Yamamoto, I. & Ohkusu, M. (2008). Advanced Technologies for Cruising AUV Urashima, *Trans. of International Journal of Offshore and Polar Engineering*, Vol. 18, No. 2, pp. 81-90, 2008.

Armand, M & Tarascon, J. M. (2008). Building better batteries, *nature*, vol. 451, no. 7, (2008) pp. 652-657.

Blandin, J. & Leon, P. (1998). Network architectures for underwater systems: two applications of the CAN bus, *Proceedings of OCEANS '98 Conference*, Vol. 1, pp. 503 - 507, 28 Sept.-1 Oct. 1998.

Bowen, A.; Yoerger, D. ; Whitcomb, L. L. & Fornari, J. (2004). Exploring the Deepest Depths : Preliminary Design of a Novel Light-Tehered Hybrid ROV for Global Science in Extreme Environments, *the Journal of the Marine Technology Society*, 38, 2,(2004) pp. 92-101.

Bradley, A. M.; Duester, A.R.; Liberatore, S.P. & Yoerger, D.R., (2000). Extending the Endurance of an Operational Scientific AUV using Lithium-Ion Batteries, *Proceedings of Unmanned Underwater Vehicle Showcase (UUVS) 2000*, Southampton, United Kingdom, September 2000.

Gornak, V.E.; Inzartsev, A.V.; Lvov, O.Yu.; Matvienko, Y.V. & Scherbatyuk, A.Ph. (2006). MMT 3000 - Small AUV of New Series of IMTP FEB RAS, *Proceedings of OCEANS 2006*, p.p. 1 – 6, ISBN: 1-4244-0115-1, Boston, MA, USA, Sept. 2006.

Hasvold, O.; Johansen, K.H. & Vestgaard, K. (2002). The alkaline aluminium hydrogen peroxide semi-fuel cell for the HUGIN 3000 autonomous underwater vehicle, *Proceedings of the 2002 Workshop on Autonomous Underwater Vehicles*, pp. 89-94, ISBN 0-7803-7572-6, Feb. 19, 2002.

Ishibashi, S.; Hyakudome, T.; Yoshida H. & Sawa, T. (2008). The Prototype To Control The Rotational Motion Applied To An Inertial Navigation System Equipped With An Autonomous Underwater Vehicle, *Proceedings of the Eighteenth (2008) International Offshore and Polar Engineering Conference*, p.p. 394-398, ISBN 1-880653-68-0, Vancouver, BC, Canada, July 6-11, 2008.

Kuperman, W. A.; Hodgkis, W. S.; Song, H. C.; Akal, T.; Ferla, C. & Jackson, D. R. (1998). Phase conjugation in the ocean: Experimental demostration of an acoustic time-reversal mirror, *J. Acoust. Soc. Am*., 103, pp. 25-40, 1998.

Larsen, M. B. (2002). High Performance Autonomous Underwater Navigation, *Hydro. International*, vol. 6, 2002.

Matsumoto, G. I. (1991). Swimming movements of ctenophores, and the mechanicsof propulsion by ctene rows, *Hydrobiologis* 216/217, pp. 319-325.

Meacham, GB Kirby (1986). Autostereoscopic Displays - Past and Future, *Advances in display technology, Proc. of SPIE*, Vol. 624.

McPhail, S. (2007). AUTOSUB6000 - a leap forward in capability: a long range and deep diving AUV for marine science, *National Oceanography Centre Southampton Informal Physics Seminar* (POETS Corner), 29 March 2007.

Murashima, T.; Nakajoh, H.; Yoshida, H.; Yamauchi, N. & Sezoko, H. (2004). 7000m class ROV KAIKO7000, *Proc. OCEANS '04 MTS/IEEE*, pp. 812 – 817,ISBN 0-7803-8669-8, Kobe, Japan, November. 9-12, 2004.

Shimura, T. (2004). A study on the focusing property of time reversal waves in shallow water, *Acoust. Sci. & Tech.* 25, 5, pp. 364-372, 2004.

Shimura, T; Yoshida, H. & Lindsay, D. (2006). Basic study on time-reversal communication in shallo water with high frequency, *Proc. of Techno-Ocean 2006 / 19th JASNAOE OES*. Paper No. 171., 2006.

Siegel, M. & King, R. W. P. (1973). Electromagnetic Propagation Between Antennas Submerged in the Ocean, *IEEE Trans. Antennas and Propagation*, AP-21,4, pp. 507-513, 1973.

Takai, K.; Inoue, A. & Horikoshi, K. (1999). Thermaerobacter marianensis gen. nov., sp. nov., an aerobic extremely thermophilic marine bacterium from the 11000 m deep Mariana Trench, *Int J Syst Bacteriol* 49 (2) pp. 619-628, 1999.

Tsukioka, S. Aoki, T. Yamamoto, I. Yoshida, H. Hyakudome, T. Sawa, T. Ishibasi, S. Mizuno, M. Tahara, J. Ishikawa, A. (2005). The PEM Fuel Cell System for Autonomous Underwater Vehicles, *the Journal of the Marine Technology Society*, 39, 3, (2005) pp. 56-64.

Walther, D.; Edgington, D.R. & Koch, C., (2004). Detection and tracking of objects in underwater video, *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. I-544- I-549 Vol.1.

Watanabe, M.; Tashiro, S. & Momma, H. (2004). Loss of the Full Ocean depth ROV Kaiko – Part 3: The Cause of Secondary Cable Fracture, *Proc. of 14th ISOPE-2004*, pp.199-202. Toulon, France, May 2004.

Watanabe, Y; Mizuno, M.; Yoshida, H. & Ochi, H. (2006). The Acoustic Positioning for The Bottom Sampler, *Proc. of Techno-Ocean 2006 / 19th JASNAOE OES*. Paper No. 179., 2006.

Weidong, L.; Li'e, G.; Yilin, D. & Jianning, X. (2006). Communication Scheduling for CAN bus Autonomous Underwater Vehicles, *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation*, pp. 379 - 383, 25-28, June 2006.

Wiebe, P. H. & Benfield, M. C. (2003). From the Hensen net toward four-dimensional biological oceanography, *Progress in Oceanography* 56, pp. 7-136.

Yoshida, H.; Lindsay, D. J.; Yamamoto, H.; Ishibashi, S.; Hyakudome, T.; Okuno, H. & Uemura, T. (2008). Development of the AUV/UROV "PICASSO". *The Proceedings of the 18th ISOPE conference*, Vol. II, p.p. 423-428, Vancouver, Canada, July 6-11, 2008.

Yoshida, H., Lindsay, D.J., Yamamoto, H., Tsukioka, S., Shimura, T., and Ishibashi S. (2007a) Small hybrid vehicles for jellyfish surveys in midwater. *The Proceedings of the 17th ISOPE conference*, p.p. 127-131, Lisbon, Lisbon, Portugal, July 1-6, 2007.

Yoshida, H.; Aoki, T.; Osawa, H.; Tsukioka, S.; Ishibashi, S.; Watanabe, Y.; Tahara, J.; Miyazaki, T.; Hyakudome, T.; Sawa, T.; Itoh, K.; Ishikawa, A. & Lindsay, D. (2007b). Newly-developed Devices for The Two Types of Underwater Vehicles, *Proceedings of OCEANS 2007 - Europe*, p.p. 1-6, ISBN: 978-1-4244-0635-7,Aberdeen, 18-21 June 2007.

Yoshida, H.; Aoki, T.; Yamamoto, I.; Tsukioka, S.; Hyakudome, T.; Ishibashi, S.; Sasamoto, R.; Nasuno, Y. & Ishikawa, A. (2004). A working AUV for scientific research, OCEANS '04. MTTS/IEEE TECHNO-OCEAN '04, Vol. 2, pp. 863 - 868, 9-12 Nov. 2004.

Young, C.; Fletcher, B.; Buescher, J.; Whitcomb, L.L.; Yoerger, D.; Bowen, A.; Mccabe, R. Heintz, M.; Fuhrmann, R.; Taylor, C. & Elder, R. (2006). Field Tests of the Hybrid Remotely Operated Vehicle (HROV) Light Fiber Optic Tether, *Proceedings of*

OCEANS 2006. MTS/IEEE, pp. 1-6, ISBN: 1-4244-0115-1, Boston, MA, USA, Sept. 2006.

Zitnick, C. & Kanade, T. (1999). A Cooperative Algorithm for Stereo Matching and Occlusion Detection, *Tech. report CMU-RI-TR-99-35, Robotics Institute*, Carnegie Mellon University, October, 1999.

Edited by Alexander V. Inzartsev

For the latest twenty to thirty years, a significant number of AUVs has been created for the solving of wide spectrum of scientific and applied tasks of ocean development and research. For the short time period the AUVs have shown the efficiency at performance of complex search and inspection works and opened a number of new important applications. Initially the information about AUVs had mainly review-advertising character but now more attention is paid to practical achievements, problems and systems technologies. AUVs are losing their prototype status and have become a fully operational, reliable and effective tool and modern multi-purpose AUVs represent the new class of underwater robotic objects with inherent tasks and practical applications, particular features of technology, systems structure and functional properties.

ISBN 978-953-51-5743-4

Photo by xalanx / iStock

IntechOpen